

NOMBRE DEL TRABAJO

DESARROLLO DE UN SISTEMA EMBEBIDO PARA LA ESTIMACIÓN DE LA FRECUENCIA RESPIRATORIA BASADO EN SEÑALES

AUTOR

Ronny Ore Alvarez

RECUENTO DE PALABRAS

15335 Words

RECUENTO DE CARACTERES

86602 Characters

RECUENTO DE PÁGINAS

84 Pages

TAMAÑO DEL ARCHIVO

3.2MB

FECHA DE ENTREGA

Feb 29, 2024 12:53 PM GMT-5

FECHA DEL INFORME

Feb 29, 2024 12:54 PM GMT-5

● 19% de similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos.

- 19% Base de datos de Internet
- Base de datos de Crossref
- 5% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

● Excluir del Reporte de Similitud

- Base de datos de trabajos entregados
- Material citado
- Coincidencia baja (menos de 8 palabras)
- Material bibliográfico
- Material citado



**FORMULARIO DE AUTORIZACIÓN PARA LA
PUBLICACIÓN DE TRABAJOS DE INVESTIGACIÓN EN
EL REPOSITORIO INSTITUCIONAL DE LA UNTELS**
(Art. 45° de la ley N° 30220 – Ley)

Autorización de la propiedad intelectual del autor para la publicación de tesis en el Repositorio Institucional de la Universidad Nacional Tecnológica de Lima Sur (<https://repositorio.untels.edu.pe>), de conformidad con el Decreto Legislativo N° 822, sobre la Ley de los Derechos de Autor, Ley N° 30035 del Repositorio Nacional Digital de Ciencia, Tecnología e Innovación de Acceso Abierto, Art. 10° del Rgto. Nacional de Trabajos de Investigación para optar grados académicos y títulos profesionales en las universidades – RENATI Res. N° 084-2022-SUNEDU/CD, publicado en El Peruano el 16 de agosto de 2022; y la RCO N° 061-2023-UNTELS del 01 marzo 2023.

TIPO DE TRABAJO DE INVESTIGACIÓN

- 1). TESIS (X) 2). TRABAJO DE SUFICIENCIA PROFESIONAL ()

DATOS PERSONALES

Apellidos y Nombres: ORE ALVAREZ, RONNY JEAN
D.N.I.: 47898624
Otro Documento:
Nacionalidad: PERUANA
Teléfono: 924530083
e-mail: 2015100947@untels.edu.pe

DATOS ACADÉMICOS

Pregrado

Facultad: FACULTAD DE INGENIERÍA Y GESTIÓN
Programa Académico: TESIS
Título Profesional otorgado: INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES

Postgrado

Universidad de Procedencia:
País:
Grado Académico otorgado:

Datos de trabajo de investigación

Título: "DESARROLLO DE UN SISTEMA EMBEBIDO PARA LA ESTIMACIÓN DE LA FRECUENCIA RESPIRATORIA BASADO EN SEÑALES DE FOTOPLETISMOGRAFÍA"
Fecha de Sustentación: 05 DE DICIEMBRE DEL 2023
Calificación: APROBADO POR UNANIMIDAD
Año de Publicación: 2024

AUTORIZACIÓN DE PUBLICACIÓN EN VERSIÓN ELECTRÓNICA

A través de la presente, autorizo la publicación del texto completo de la tesis, en el Repositorio Institucional de la UNTELS especificando los siguientes términos:

Marcar con una X su elección.

- 1) Usted otorga una licencia especial para publicación de obras en el REPOSITORIO INSTITUCIONAL DE LA UNIVERSIDAD NACIONAL TECNOLÓGICA DE LIMA SUR.

Si autorizo No autorizo

- 2) Usted autoriza para que la obra sea puesta a disposición del público conservando los derechos de autor y para ello se elige el siguiente tipo de acceso.

Derechos de autor		
TIPO DE ACCESO	ATRIBUCIONES DE ACCESO	ELECCIÓN
ACCESO ABIERTO 12.1(*)	info:eu-repo/semantics/openAccess (Para documentos en acceso abierto)	(X)

- 3) Si usted dispone de una **PATENTE** puede elegir el tipo de **ACCESO RESTRINGIDO** como derecho de autor y en el marco de confiabilidad dispuesto por los numerales 5.2 y 6.7 de la directiva N° 004-2016-CONCYTEC DEGC que regula el Repositorio Nacional Digital de CONCYTEC (Se colgará únicamente datos del autor y el resumen del trabajo de investigación).

Derechos de autor		
TIPO DE ACCESO	ATRIBUCIONES DE ACCESO	ELECCIÓN
ACCESO RESTRINGIDO	info:eu-repo/semantics/restrictedAccess (Para documentos restringidos)	()
	info:eu-repo/semantics/embargoedAccess (Para documentos con períodos de embargo. Se debe especificar las fechas de embargo)	()
	info:eu-repo/semantics/closedAccess (para documentos confidenciales)	()

(*) <http://renati.sunedu.gob.pe>



UNIVERSIDAD NACIONAL
TECNOLÓGICA DE LIMA SUR

Rellene la siguiente información si su trabajo de investigación es de acceso restringido:

Atribuciones de acceso restringido:

Motivos de la elección del acceso restringido:

ORE ALVAREZ, RONNY JEAN

APELLIDOS Y NOMBRES

47898624

DNI

Firma y huella:



Lima, 29 de FEBRERO del 2024

UNIVERSIDAD NACIONAL TECNOLÓGICA DE LIMA SUR

**FACULTAD DE INGENIERÍA Y GESTIÓN
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA Y
TELECOMUNICACIONES**



**“DESARROLLO DE UN SISTEMA EMBEBIDO PARA LA ESTIMACIÓN
DE LA FRECUENCIA RESPIRATORIA BASADO EN SEÑALES DE
FOTOPLETISMOGRAFÍA”**

TESIS

Para optar el Título Profesional de

INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES

PRESENTADO POR EL BACHILLER

ORE ALVAREZ, RONNY JEAN
ORCID: 0009-0006-8553-5857

ASESOR
LEZAMA CALVO, JINMI GREGORY
ORCID: 0000-0003-2906-9741

**Villa El Salvador
2023**



DECANATO DE LA FACULTAD DE INGENIERÍA Y GESTIÓN

ACTA DE SUSTENTACIÓN DE TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE
INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES

En Villa El Salvador, siendo las **10:40** horas del día 5 de diciembre de 2023, en la Facultad de Ingeniería y Gestión, los miembros del Jurado Evaluador, integrado por:

PRESIDENTE: DR. MARK DONNY CLEMENTE ARENAS DNI N° 41962207 C.I.P. N° 181400
SECRETARIO: MG. PABLO ANDRÉS VILLEGAS CHUNGA DNI N° 09694556 C.I.P. N° 199274
VOCAL : MG. EDGARD OPORTO DÍAZ DNI N° 09352077 C.I.P. N° 106881

ASESOR : DR. JINMI GREGORY LEZAMA CALVO DNI N° 42294872 C.I.P. N° 97712

Designados mediante Resolución de Decanato N° 338-2023-UNTELS-R-D de fecha 15 de agosto de 2023 quienes dan inicio a la Sesión Pública de Sustentación y Evaluación de Tesis.

Acto seguido, el (la) aspirante al : Grado de Bachiller Título Profesional

Don: RONNY JEAN ORE ALVAREZ identificado(a) con D.N.I. N° 47898624 procedió a la Sustentación de:

Trabajo de investigación Tesis Trabajo de suficiencia Artículo científico

Titulada: "DESARROLLO DE UN SISTEMA EMBEBIDO PARA LA ESTIMACIÓN DE LA FRECUENCIA RESPIRATORIA BASADO EN SEÑALES DE FOTOPLETISMOGRAFÍA".

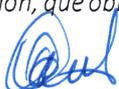
Aprobado mediante Resolución de Decanato N° 965-2023-UNTELS-R-D de fecha 27 de noviembre de 2023, de conformidad con las disposiciones del Reglamento General de Grados Académicos y Títulos Profesionales vigentes, sustentó y absolvió las interrogantes que le formularon los señores miembros del Jurado Evaluador.

Concluida la Sustentación se procedió a la evaluación y calificación correspondiente, resultando el aspirante APROBADO por **unanimidad** con la nota de: **trece**(letras)..... **13** (números), de acuerdo al Art. 65° del Reglamento General para optar el Título Profesional.

CALIFICACIÓN		CONDICIÓN	EQUIVALENCIA
NÚMERO	LETRAS		
13	trece	Aprobado por unanimidad	Bueno

Siendo las **11:10** horas del día 5 de diciembre de 2023, se dio por concluido el acto de sustentación, firmando el jurado evaluador el Acta de Sustentación, que obra en el Decanato de la Facultad de Ingeniería y Gestión.


MG. PABLO ANDRÉS VILLEGAS CHUNGA
SECRETARIO


DR. MARK DONNY CLEMENTE ARENAS
PRESIDENTE


MG. EDGARD OPORTO DÍAZ
VOCAL


RONNY JEAN ORE ALVAREZ
BACHILLER

DEDICATORIA

El presente trabajo de investigación se lo dedico a mi madre, que me enseñó a ser fuerte frente a las adversidades y no rendirse.

A mi padre, que es un ejemplo de esfuerzo y sacrificio.
A mis hermanos, Tonny y Yomar, que fueron mi ejemplo a seguir.

A todos y cada uno de los que me apoyaron incondicionalmente.

AGRADECIMIENTOS

Quiero expresar mi más sincero agradecimiento por el inquebrantable apoyo y orientación que me han brindado durante el tiempo que dedicamos a mi trabajo de investigación y tesis. Este logro no habría sido posible sin su invalorable contribución.

A mi Asesor, Dr. Jinmi Lezama Calvo:

Agradezco profundamente su guía experta, paciencia y dedicación a lo largo de este viaje académico. Su sabiduría y orientación han sido fundamentales para dar forma a mis ideas y convertirlas en una tesis que refleje no solo mi crecimiento académico sino también la excelencia que usted inspira.

A mis Revisores, Mg. Pablo Villegas Chunga y Mg. Edgard Oporto Díaz:

Quiero expresar mi gratitud por su compromiso y tiempo dedicado a revisar y perfeccionar mi trabajo. Sus valiosas sugerencias y observaciones han enriquecido significativamente mi investigación, brindándome la oportunidad de mejorar y perfeccionar mi contribución académica.

A mi Querida Familia:

Mi más profundo agradecimiento va para ustedes, mi familia, por el amor, el apoyo incondicional y la comprensión que me han brindado en cada paso de este proceso. Sus palabras alentadoras y presencia constante han sido mi fuente de fortaleza, inspirándome a esforzarme siempre por alcanzar mis metas.

Este logro no solo es mío, sino de todos ustedes que han compartido este viaje conmigo. Cada desafío superado y cada éxito alcanzado ha sido posible gracias a su respaldo constante.

RESUMEN

En la presente investigación se propone desarrollar un hardware embebido portable para la adquisición de señales de fotopletismografía, para su posterior procesamiento mediante un algoritmo que permita estimar la frecuencia respiratoria. En primer lugar, se diseñó el hardware, para ello se eligió el sensor MAX30102 que es un pulsioxímetro comercial, la lectura de data se hizo mediante la placa Arduino Pro Mini que incluye un microcontrolador ATmega328p, la comunicación se realizó a través del módulo Bluetooth HC-05 y para la alimentación se utilizó una batería de Ion Litio y un módulo cargador TP4056. Posteriormente, se obtuvieron señales PPG con data cruda de voluntarios de prueba. Para el procesamiento de estas señales se siguió un algoritmo de 4 etapas: filtrado de la señal, detección de picos de la señal filtrada, cálculo del intervalo de serie temporal de picos y valles, y por último la estimación de la frecuencia respiratoria. Para el filtrado se utilizó un filtro Butterworth pasa bandas de tercer orden. El algoritmo utilizado logro buenos resultados en comparación a otros algoritmos utilizados en la literatura, se obtuvo un error medio absoluto de 0.653 respiraciones por minuto en un conjunto de 20 muestras.

Palabras claves: PPG, MAX30102, Microcontrolador, Frecuencia Respiratoria

ABSTRACT

The present research aims to develop a portable embedded hardware for the acquisition of photoplethysmography (PPG) signals, with subsequent processing through an algorithm to estimate respiratory frequency. First, the hardware was designed, choosing the MAX30102 sensor, a commercial pulse oximeter. Data reading was performed using the Arduino Pro Mini board, which includes an ATmega328p microcontroller. Communication was established through the HC-05 Bluetooth module, and a Lithium Ion battery along with a TP4056 charging module were used for power supply. Subsequently, PPG signals with raw data were obtained from test volunteers. For signal processing, a four-stage algorithm was employed: signal filtering, detection of peaks in the filtered signal, calculation of the time-series interval between peaks and valleys, and finally, the estimation of respiratory frequency. A third-order Butterworth bandpass filter was used for filtering. The algorithm yielded favorable results compared to other algorithms found in the literature, achieving a mean absolute error of 0.653 breaths per minute in a set of 20 samples.

Keywords: PPG, MAX30102, Microcontroller, Respiratory Rate

ÍNDICE

DEDICATORIA	ii
AGRADECIMIENTOS.....	iii
RESUMEN	iv
ABSTRACT	v
ÍNDICE.....	vi
LISTA DE FIGURAS.....	viii
LISTA DE TABLAS.....	x
INTRODUCCIÓN	11
I. PLANTEAMIENTO DEL PROBLEMA.....	12
1.1. Motivación.....	12
1.2. Estado del arte.....	12
1.3. Descripción del problema	13
1.4. Formulación del problema	14
1.4.1. <i>Problema general</i>	14
1.4.2. <i>Problemas específicos</i>	14
1.5. Objetivos.....	14
1.5.1. <i>Objetivo general</i>	14
1.5.2. <i>Objetivos específicos</i>	14
1.6. Justificación del problema	15
II. MARCO TEÓRICO.....	16
2.1. Antecedentes de la investigación.....	16
2.1.1. <i>Nacionales</i>	16
2.1.2. <i>Internacionales</i>	17
2.2. Bases teóricas.....	18
2.2.1. <i>Signos vitales</i>	18
2.2.2. <i>Sistemas electrónicos</i>	25
2.2.3. <i>Procesamiento digital de señales</i>	33
III. METODOLOGÍA.....	39
3.1. Descripción de la metodología.....	39
3.2. Implementación de la investigación.....	39
3.2.1. <i>Desarrollo de Hardware</i>	39
3.2.2. <i>Algoritmo Propuesto</i>	45

3.2.3.	<i>Desarrollo de dashboard.....</i>	54
3.2.4.	<i>Validación de prototipo.....</i>	57
3.3.	Resultados.....	58
3.3.1.	<i>Hardware y adquisición de data</i>	58
3.3.2.	<i>Algoritmo, dashboard y validación</i>	60
IV.	DISCUSIÓN DE RESULTADOS	66
V.	CONCLUSIONES	67
VI.	REFERENCIAS BIBLIOGRÁFICAS	68
ANEXOS.....		72
Anexo 1.	Matriz de Consistencia.....	72
Anexo 2.	Glosario de términos	73
Anexo 3.	Código en Arduino para la comunicación y configuración del sensor MAX30102	74
Anexo 4.	Código en MATLAB para la lectura y representación de datos del sensor	76
Anexo 5.	Código en MATLAB para filtrar la señal PPG	77
Anexo 6.	Código en MATLAB para la ubicación de picos y valles.....	78
Anexo 7.	Código en MATLAB para el cálculo de la Frecuencia Respiratoria	79
Anexo 8.	Código en App Designer de MATLAB para el desarrollo del Dashboard	80

LISTA DE FIGURAS

Figura 1 <i>Ciclo de la Señal PPG</i>	19
Figura 2 <i>Espectros de absorción de la hemoglobina.</i>	20
Figura 3 <i>Modo Transmisivo</i>	21
Figura 4 <i>Modo Reflexivo</i>	21
Figura 5 <i>Componentes de la señal PPG</i>	22
Figura 6 <i>Calidad de la Señal PPG</i>	23
Figura 7 <i>Diagrama Funcional de MAX30102</i>	25
Figura 8 <i>Módulo MAX30102</i>	26
Figura 9 <i>Pines del módulo MAX30102</i>	26
Figura 10 <i>Pines Arduino Pro Mini</i>	28
Figura 11 <i>Distribución de pines del módulo Bluetooth HC-05</i>	30
Figura 12 <i>Batería Ion Litio</i>	31
Figura 13 <i>Detalle de pines de conexión del módulo cargador TP4056</i>	32
Figura 14 <i>Diagrama de bloques del hardware</i>	39
Figura 15 <i>Diagrama de conexión del dispositivo</i>	40
Figura 16 <i>Prototipo final</i>	41
Figura 17 <i>Diagrama de flujo del código en Arduino</i>	42
Figura 18 <i>Gráfica de los datos ADC recibidos en Matlab</i>	44
Figura 19 <i>Acercamiento de la señal PPG</i>	45
Figura 20 <i>Pasos del algoritmo propuesto</i>	46
Figura 21 <i>Transformada rápida de Fourier de la señal infrarroja</i>	46
Figura 22 <i>Acercamiento de la Transformada rápida de Fourier de la señal infrarroja</i> ...	47
Figura 23 <i>Código de diseño de filtro en MATLAB</i>	49
Figura 24 <i>Gráfica de respuesta del filtro en MATLAB</i>	49
Figura 25 <i>Señal PPG del LED infrarrojo filtrada</i>	50
Figura 26 <i>Transformada de Fourier de la señal PPG filtrada del LED infrarrojo</i>	50
Figura 27 <i>Parámetros de la función 'findpeaks'</i>	51
Figura 28 <i>Picos y valles de la señal filtrada</i>	52
Figura 29 <i>Tiempo correspondiente a las ubicaciones de los picos y valles</i>	53
Figura 30 <i>Interfaz gráfica MATLAB botón Desing App</i>	55
Figura 31 <i>Ventana de entorno de App Designer</i>	55
Figura 32 <i>Ventana de entorno de App Designer</i>	56

Figura 33 <i>Dashboard para el cálculo de Frecuencia Respiratoria</i>	57
Figura 34 <i>Lado inferior del prototipo</i>	58
Figura 35 <i>Lado frontal y superior del prototipo</i>	59
Figura 36 <i>Toma de muestras con el prototipo</i>	59
Figura 37 <i>Dashboard del sujeto 1.</i>	60
Figura 38 <i>Dashboard del sujeto 2.</i>	61
Figura 39 <i>Dashboard del sujeto 3.</i>	62
Figura 40 <i>Dashboard del sujeto 4.</i>	63

LISTA DE TABLAS

Tabla 1 <i>Valores normales de FR en RPM</i>	24
Tabla 2 <i>Descripción de pines del módulo MAX30102</i>	27
Tabla 3 <i>Especificaciones técnicas del módulo MAX30102</i>	27
Tabla 4 <i>Pines con funciones especializadas</i>	28
Tabla 5 <i>Especificaciones técnicas ATmega328P</i>	29
Tabla 6 <i>Función de pines del módulo Bluetooth HC-05</i>	30
Tabla 7 <i>Especificaciones técnicas del módulo Bluetooth HC-5</i>	31
Tabla 8 <i>Especificaciones técnicas de Batería Ion Litio</i>	32
Tabla 9 <i>Especificaciones técnicas del Cargador TP4056</i>	33
Tabla 10 <i>Funciones de ventana para filtros FIR</i>	35
Tabla 11 <i>Comparación de ventanas comúnmente utilizadas</i>	36
Tabla 12 <i>Valores ADC del LED Infrarrojo y LED Rojo</i>	43
Tabla 13 <i>Resultados obtenidos del sujeto 1</i>	60
Tabla 14 <i>Resultados obtenidos del sujeto 2</i>	61
Tabla 15 <i>Resultados obtenidos del sujeto 3</i>	62
Tabla 16 <i>Resultados obtenidos del sujeto 4</i>	63
Tabla 17 <i>Resultados aproximados de estimación de FR</i>	64
Tabla 18 <i>Resultados de estimación de FR</i>	64

INTRODUCCIÓN

La oximetría de pulso o la pulsioximetría es un método no invasivo basado en la fotopleletismografía (PPG) que se utiliza para medir el porcentaje de saturación de oxígeno en la sangre (SpO_2), como también la frecuencia cardíaca, este método se ha vuelto recurrente a la hora de monitorear pacientes anestesiados y a aquellos que se encuentran en cuidados intensivos (UCI).

Dado su modo simple de operación y la comodidad que brinda a los pacientes, se ha planteado la exploración de medidas adicionales que podamos derivar del pulsioxímetro, uno de ellos es la obtención de la frecuencia respiratoria.

La frecuencia respiratoria es un antecedente conocido de muchos episodios de deterioro fisiológico en pacientes, es importante pues tiene muchos usos clínicos, y sus mediciones están indicadas en muchos entornos de cuidados intensivos y quirúrgicos.

Actualmente en la medición de la frecuencia respiratoria se utiliza un capnógrafo para monitorear la producción de CO_2 , dicho dispositivo es muy costoso y requiere de un alto grado de mantenimiento. Además de ello, resulta incómodo para los pacientes pues hace uso de una máscara o una cánula nasal. Dicho esto, es importante obtener una frecuencia respiratoria precisa, porque esta información permite caracterizar si una lectura de baja saturación de oxígeno se debe a bajas tasas de respiración o es el resultado de otras condiciones fisiológicas peligrosas.

La comodidad del paciente también mejora enormemente al obtener esta información adicional del oxímetro de pulso sin tener que usar un dispositivo adicional para medir la frecuencia respiratoria.

Esta investigación busca desarrollar un hardware de bajo costo con electrónica embebida y que pueda usarse como una herramienta para la estimación de frecuencia respiratoria.

I. PLANTEAMIENTO DEL PROBLEMA

1.1. Motivación

La frecuencia respiratoria (FR) es un indicador de salud ampliamente utilizado junto con otros parámetros para evaluar la salud de los pacientes en hospitales y clínicas. Sin embargo, la FR se suele registrar mediante técnicas como la espirometría o la neumografía. Estas técnicas requieren uso de dispositivos o monitores que pueden interferir con la respiración natural del paciente, siendo inviables bajo ciertas condiciones como, por ejemplo, monitorización ambulatoria, estudios del sueño o pruebas de esfuerzo. Por consiguiente, es necesario proporcionar una solución alternativa desarrollando un método electrónico automatizado para medir la FR utilizando un dispositivo no invasivo.

1.2. Estado del arte

Para la extracción de componentes respiratorios se han considerado otras variaciones inducidas por la respiración en la señal de fotopletimografía (PPG, por sus siglas en inglés). Estos se basan en la extracción de características adicionales de la señal. Muchas técnicas exploran de diversas maneras una o más de las variaciones inducidas por la respiración. El método propuesto por, M. Pirhonen et al., (2018) consiste en utilizar la PPG para extraer la sobrecarga de amplitud respiratoria (RIAV, por sus siglas en inglés) de la señal y luego aplicar un filtro de partículas para estimar la FR. D. Jarchi et al., (2019) propone un método que consiste en utilizar la PPG para estimar la FR durante el movimiento, para lograr esto, se reconstruye una señal PPG limpia eliminando los artefactos de movimiento mediante un algoritmo de filtrado adaptativo y luego se utiliza esta señal reconstruida para estimar la FR utilizando un algoritmo de detección de picos. D. Bian et al., (2020) propone un enfoque que consiste en entrenar un modelo de aprendizaje profundo basado en una red neuronal convolucional (CNN, por sus siglas en inglés) llamada ResNet. Esta arquitectura de CNN utiliza conexiones de atajo para mejorar la optimización y el rendimiento del modelo. Se utiliza la señal de PPG como entrada para estimar la FR como medida de salida. En una investigación más reciente, T. Iqbal et al., (2022), propone un nuevo algoritmo para estimar la frecuencia respiratoria utilizando una señal de sensor de PPG para el seguimiento de la salud. El algoritmo es resistente a la pérdida de señal y puede manejar señales de baja calidad del sensor. Combina ventanas selectivas,

preprocesamiento y acondicionamiento de señales, filtrado Welch modificado y postprocesamiento para lograr alta precisión y robustez al ruido.

En este estudio, se buscará la aplicación del algoritmo más eficiente para la estimación de la frecuencia respiratoria, basándonos en los estudios más recientes.

1.3. Descripción del problema

Actualmente, las enfermedades respiratorias; como las infecciones respiratorias agudas (IRA), son muy comunes en todo el mundo y representan una importante causa de muerte. La neumonía es la infección respiratoria grave más frecuente, representa el 14% de todas las muertes en niños menores de 5 años y en 2019 causó la muerte de 740 180 niños (Organización Mundial de la Salud: OMS, 2022). Otra de las enfermedades más comunes es la enfermedad pulmonar obstructiva crónica (EPOC), que es la tercera causa de muerte en el mundo, en 2019 provocó 3.23 millones de muertes (Organización Mundial de la Salud: OMS, 2023). Muchas veces estas enfermedades requieren de hospitalización, y posteriormente rehabilitación.

La frecuencia respiratoria (FR) es un indicador ampliamente utilizado junto con otros parámetros, como la frecuencia cardiaca (FC) y la saturación de oxígeno en la sangre (SpO_2), para monitorear la salud de los pacientes con enfermedades respiratorias en hospitales y clínicas. Sin embargo, mientras que la FC y la SpO_2 se puede medir de forma continua, utilizando oximetría de pulso, la estimación continua de FR se basa en el uso de equipos adicionales, como el capnógrafo o el respirómetro, estos equipos resultan costosos y requieren un alto grado de mantenimiento, además, resultan incómodos y generan dificultades para el seguimiento cuando el paciente está fuera del centro de atención, ya que no permiten continuar con el monitoreo, comprometiendo así la salud del paciente.

Por lo tanto, un método preventivo en estos casos, es el seguimiento ambulatorio de los pacientes mediante sensores portátiles no molestos y livianos, capaces de registrar información de manera digital que facilitaría el tratamiento posterior del paciente.

Si bien, actualmente existen dispositivos como los relojes inteligentes, capaces de medir estos parámetros de forma casi precisa, la disposición de la data registrada es limitada por parte del usuario, por consiguiente, no puede ser usada para

un diagnóstico médico oportuno, así pues, terminan siendo un dispositivo más para complementar una línea de gadgets tecnológicos.

Por estas razones se pretende desarrollar un prototipo de pulsioxímetro económico con electrónica embebida, que pueda utilizarse como una herramienta de detección multifuncional para la detección de anomalías respiratorias.

1.4. Formulación del problema

1.4.1. Problema general

La estimación continua de la FR en pacientes de hospitales y clínicas se fundamenta en la utilización de equipos que se manipulan en los centros de atención médica y constituyen dificultades para la supervisión cuando el paciente esta fuera del establecimiento de salud.

1.4.2. Problemas específicos

- La falta de un dispositivo u oxímetro de pulso que funcione como monitor de signos vitales multipropósito.
- La falta de solidez en los métodos existentes para la estimación robusta de FR a partir de señales de PPG.
- La necesidad de sistemas que permitan visualizar datos de la FR.
- La necesidad de verificar la precisión de los prototipos existentes.

1.5. Objetivos

1.5.1. Objetivo general

Desarrollar un sistema embebido para la estimación de frecuencia respiratoria basado en señales de fotopletismografía.

1.5.2. Objetivos específicos

- Desarrollar un hardware no invasivo portable, inalámbrico para la adquisición de señales de fotopletismografía.
- Determinar una técnica para la estimación de la frecuencia respiratoria.
- Diseñar un sistema que permita visualizar los datos en un dashboard.
- Validar el prototipo mediante pruebas en laboratorio.

1.6. Justificación del problema

Las enfermedades respiratorias, son muy frecuentes en todo el mundo, esto hace que las personas con deficiencias fisiológicas en este aspecto, sean más propensas a enfermar de manera más severa, llegando a ser hospitalizadas e incluso perecer ante estas enfermedades.

Asimismo, cuando nos trasladamos a lugares de mucha altitud donde el aire se vuelve más delgado y hay menos oxígeno disponible, existe la necesidad de evaluar con premura las deficiencias fisiológicas que podamos presentar, más aún cuando los centros de salud son escasos y posiblemente sean de difícil acceso.

Por ello, es muy importante la implementación de un prototipo de pulsioxímetro capaz de monitorear de la forma más precisa indicadores tales como la frecuencia respiratoria, ya que ayuden a prever posibles complicaciones en el futuro proporcionando un control diario fuera del establecimiento de salud, así también brindando una mayor comodidad al paciente, y mejorando su calidad de vida. Además, permite al especialista médico hacer una mejor evaluación y dar un mejor tratamiento.

II. MARCO TEÓRICO

2.1. Antecedentes de la investigación

2.1.1. Nacionales

(Chávez y Guardia, 2019) en su tesis de pregrado “Aplicación de inteligencia artificial en un sistema de análisis en tiempo real de los signos vitales para pacientes con riesgo cardiaco del hospital Centro Médico Naval”, tuvo como objetivo aplicar la inteligencia artificial en un sistema de análisis en tiempo real de los signos vitales para pacientes con riesgo cardiaco del Hospital Centro Médico Naval. El sistema utilizó un dispositivo electrónico, que evaluó constantemente los indicadores vitales fundamentales (SpO_2 , FC y FR). Se usó la lógica difusa para analizar los datos obtenidos con el fin de obtener una respuesta más exacta en la interpretación de la información. Como resultado se obtuvo un sistema portátil, el cual paso pruebas de conectividad sin perder datos y monitoreando continuamente al paciente.

(Alcorta y Pinedo, 2020) en su tesis de pregrado “Desarrollo de un sistema electrónico biomédico portátil para el monitoreo y registro de signos vitales de las personas en la ciudad de Trujillo - Perú”, tuvo como objetivo desarrollar un sistema electrónico biomédico portátil, para el monitoreo y registro histórico de signos vitales de las personas en la ciudad de Trujillo - Perú en el año 2020. Para ello se realizó un procesamiento y evaluación de los datos de, temperatura corporal (TC), FC y SpO_2 obtenidos con el prototipo. Los resultados demostraron que mediante el desarrollo de un sistema electrónico biomédico portátil, es posible mejorar la forma de monitoreo y registro de los signos vitales de las personas.

(Lizarzaburu y Villanueva, 2023) en su tesis de pregrado “Desarrollo de un filtro para señales fotopleletismográficas (PPG) obtenidas en una tarjeta de adquisición de datos en un entorno de laboratorio”, tuvo como objetivo desarrollar un filtro digital para señales PPG obtenidas de una Tarjeta de Adquisición de Datos (TAD) en un entorno de laboratorio. Para ello se utilizaron señales PPG sin filtrar y mediante ecuaciones y algoritmos se compararon distintos filtros y se determinó el más eficiente. Los resultados determinaron que el filtro Butterworth de orden 6 es el filtro más apropiado para implementarse en un microcontrolador de una TAD.

2.1.2. *Internacionales*

(M. Pirhonen et al., 2018) El objetivo de este estudio fue proponer un método para estimar la FR utilizando señales PPG y evaluar la precisión de la estimación de FR utilizando un filtro de partículas. Se estudió el uso de la variabilidad de amplitud de la señal PPG de dedo en modo de transmitancia en la estimación de FR mediante la comparación de cuatro métodos de representación de frecuencia de tiempo (TF) de la señal en cascada con un filtro de partículas. Los métodos TF comparados fueron la transformada de Fourier de tiempo corto (STFT, por sus siglas en inglés) y tres tipos de métodos de sincronización sincronizada. Los resultados indicaron que los métodos avanzados de reasignación de frecuencias pueden presentar mejoras sobre los métodos lineales, como STFT. Los mejores resultados se lograron utilizando la transformada de sincronización de ondículas (WSST, por sus siglas en inglés), con un error absoluto medio y un error mediano de 2,33 y 1,15 respiraciones por minuto, respectivamente. Los métodos de compresión sincronizada fueron generalmente más precisos que STFT en la mayoría de los sujetos cuando se aplicó el filtrado de partículas.

(D. Jarchi et al., 2019) El objetivo de este trabajo fue proponer una nueva técnica que realiza la reducción de movimiento de las señales PPG con la ayuda de señales de aceleración simultáneas donde los sensores PPG y acelerómetro deben estar integrados en la misma unidad de sensor. Se usó una técnica basada en autorregresión (AR) para estimar la FR a partir de PPG reconstruidos. El método propuesto ha proporcionado resultados prometedores para la estimación de FR y sus variaciones a partir de señales PPG corruptas con artefactos de movimiento. La plataforma propuesta puede contribuir al monitoreo continuo en el hospital y en el hogar de pacientes que utilizan señales PPG en diversas condiciones, como estados de reposo y movimiento.

(D. Bian et al., 2020) En esta investigación se propuso un modelo de aprendizaje profundo para calcular la FR a partir de señales PPG y se comparó su desempeño con un método tradicional de cálculo de FR. Se propone un enfoque de aprendizaje profundo de extremo a extremo basado en la arquitectura de red residual (ResNet). Este enfoque toma datos de PPG de series temporales como entrada, aprende las reglas a través del proceso de capacitación que involucró un conjunto de datos de PPG sintético adicional generado para superar el problema de datos insuficientes del aprendizaje profundo y proporciona una estimación de FR como

salida. Los resultados revelaron que el modelo de aprendizaje profundo logró un rendimiento comparable al de un método clásico.

(T. Iqbal et al., 2022) El propósito de esta investigación fue desarrollar un algoritmo para estimar la frecuencia respiratoria a partir de la señal de PPG recopilada a través de un oxímetro de pulso, con el fin de proporcionar una solución simple y económica. Se utilizó el método de Transformada Rápida de Fourier (FFT) para analizar la señal PPG y extraer la información relacionada con la frecuencia respiratoria. Además, se aplicó un método de escalamiento para ajustar los valores estimados de la frecuencia respiratoria dentro de un umbral específico y evitar valores que excedan la frecuencia respiratoria fisiológicamente posible. El algoritmo propuesto demostró ser preciso y exacto en la estimación de la frecuencia respiratoria, superando a otros algoritmos existentes en términos de rendimiento.

2.2. Bases teóricas

2.2.1. Signos vitales

2.2.1.1. Fotopletismografía (PPG)

McDuff (2014) la define como una técnica no invasiva y de bajo costo que se puede utilizar para detectar cambios en el volumen sanguíneo cardiovascular mediante la luz transmitida o reflejada desde la superficie de la piel.

2.2.1.2. Señal PPG

La señal PPG refleja el flujo de la sangre en una arteria: la sangre viaja desde el corazón hacia las partes distales de las manos y los pies por los vasos sanguíneos en un movimiento ondulatorio, esta señal cuenta con los siguientes parámetros:

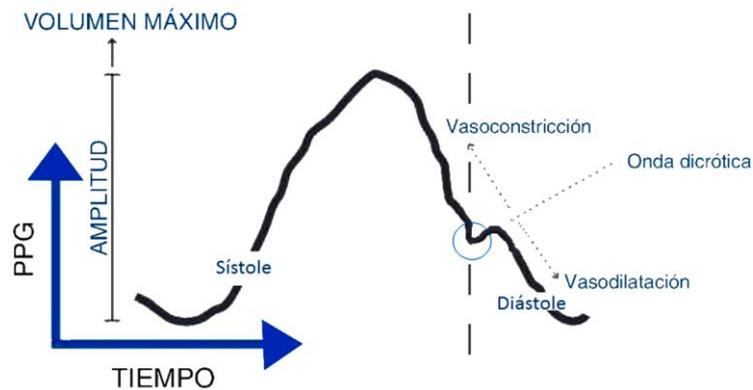
- a. Vasoconstricción.** Este proceso implica el estrechamiento o reducción del diámetro de los vasos sanguíneos, lo que eleva la resistencia al flujo de la sangre por ellos.
- b. Vasodilatación.** Este proceso consiste en el ensanchamiento o dilatación de los vasos sanguíneos, debido a que el músculo se relaja, lo que facilita el flujo de sangre hacia los diferentes órganos y tejidos del organismo.
- c. Onda dicrótica.** Indica el fin de la sístole con el cierre de la válvula aórtica, lo que provoca un incremento breve en el volumen de sangre arterial (Cometti y Vogel, 2018).

- d. Amplitud Sistólica.** Es el pico máximo de volumen que se registra en la zona sometida a medición correspondiente a la fase sistólica del corazón, donde se expulsa la sangre (Cometti y Vogel, 2018).
- e. Amplitud Diastólica.** Es la presión que la sangre ejerce cuando el corazón se distiende para volver a llenarse de sangre (Cometti y Vogel, 2018).

En la Figura 1 se puede observar gráficamente los parámetros descritos en un ciclo de la señal PPG.

Figura 1

Ciclo de la Señal PPG



Nota. La figura muestra el ciclo completo de la señal PPG de un individuo sano.

Fuente: Cometti y Vogel (2018).

2.2.1.2.1. Adquisición de la señal PPG

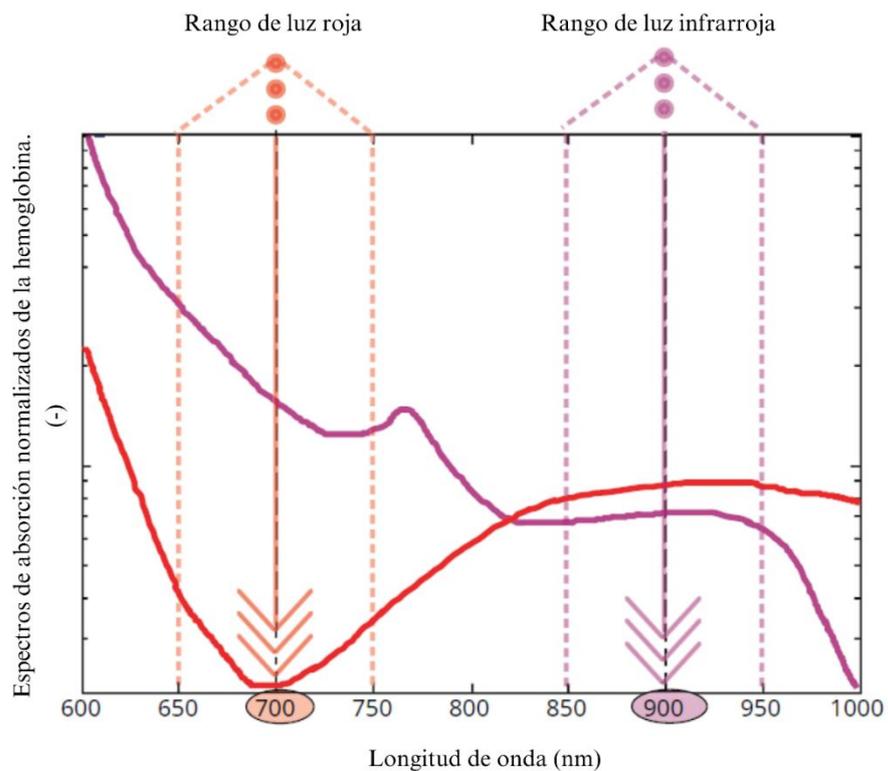
Según Peralta (2015), para medir la señal PPG se requiere una fuente de luz que ilumine el tejido que se quiere analizar y un detector óptico que registre las variaciones en la intensidad de luz recibida, que son proporcionales a los cambios de volumen de sangre. Las señales se obtienen para dos longitudes de onda, en el rango del rojo (650-750 nm) y del infrarrojo (850-950 nm).

La señal roja se usa para medir la frecuencia cardiaca, que es el número de latidos del corazón por minuto esta se refleja más en la sangre poco oxigenada, que es la que circula por las venas. Cuando el corazón late, se produce un aumento del flujo sanguíneo en las venas, lo que se traduce en un aumento de la señal roja reflejada, mientras que la señal infrarroja se usa para medir la saturación de

oxígeno en la sangre, que es el porcentaje de hemoglobina que transporta oxígeno en los glóbulos rojos. La señal infrarroja se refleja más en la sangre oxigenada, que es la que circula por las arterias. La saturación de oxígeno varía según la respiración y la actividad física. En la Figura 2 se muestra dos curvas que representan los espectros de absorción de la hemoglobina.

Figura 2

Espectros de absorción de la hemoglobina.



Nota. La curva roja representa la hemoglobina oxigenada, mientras que la curva morada representa la hemoglobina desoxigenada. Fuente: Adaptado de Elgendi (2021).

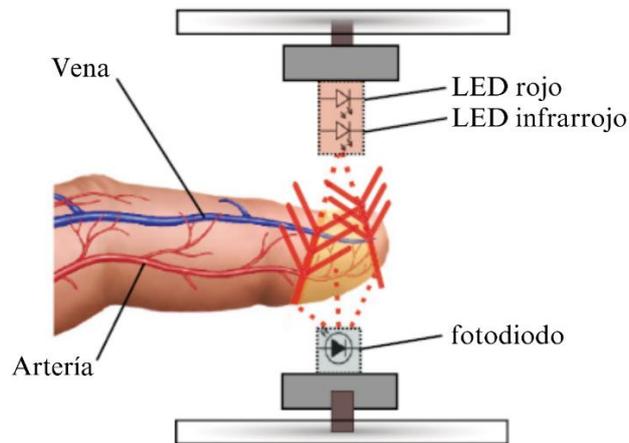
Según la disposición de los emisores y el fotodetector. Existen dos modos básicos de configuración:

a. Modo Transmisivo (por transmisión de la luz)

En el modo transmisivo, como se muestra en la Figura 3, el emisor y el fotodetector están uno frente al otro, el sitio de medición se encuentra en la entre ambos, es decir la luz transmitida a través del medio es detectado por un fotodetector opuesto a la fuente LED.

Figura 3

Modo Transmisivo



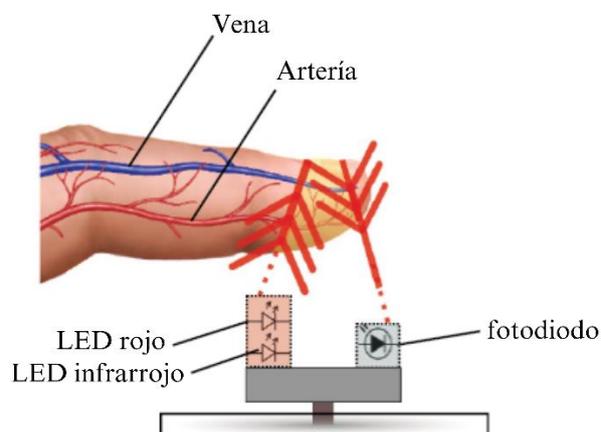
Nota. La luz transmitida a través del medio es detectada por un fotodiodo opuesto a las fuentes LED. Fuente: Elgendi (2021)

b. Modo Reflexivo (por reflexión de la luz)

En el modo reflexivo, como se muestra en la Figura 4, el emisor y el fotodetector se encuentran uno al lado del otro, encima del lugar de medición. La luz rebota desde el emisor al detector a través del sitio de medición, es decir el fotodetector detecta la luz que se retro dispersa o se refleja en el tejido, el hueso, y/o vasos sanguíneos.

Figura 4

Modo Reflexivo



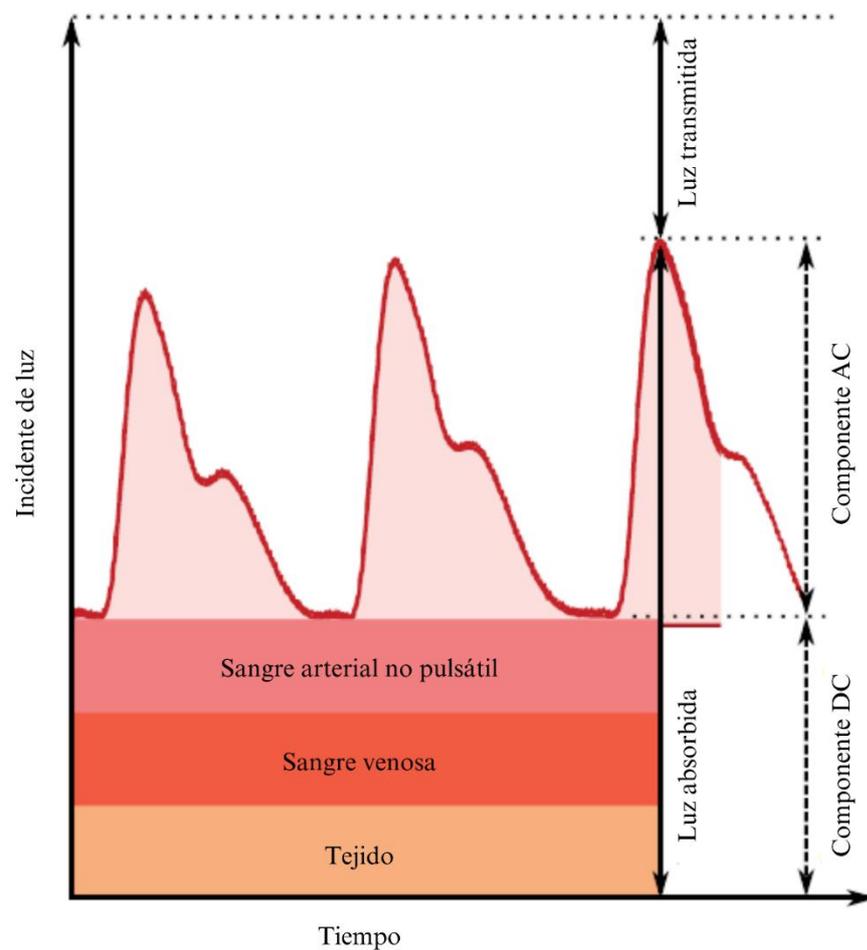
Nota. El fotodiodo detecta la luz reflejada desde el tejido, hueso y/o vasos sanguíneos. Fuente: Elgendi (2021)

2.2.1.2.2. Morfología de la señal PPG

Según Allen (2007), la señal PPG se compone de una onda pulsátil que varía con cada latido del corazón (AC), debido a los cambios en el volumen de sangre causados por el bombeo cardíaco, y de una línea de base que cambia lentamente (DC), relativa a la atenuación que producen los tejidos subyacentes que envuelven las arterias, como se observa la Figura 5.

Figura 5

Componentes de la señal PPG



Nota. La figura muestra la transmisión y absorción de luz dentro del tejido, así como las componentes de la señal PPG. Fuente: Adaptado de Elgendi (2021).

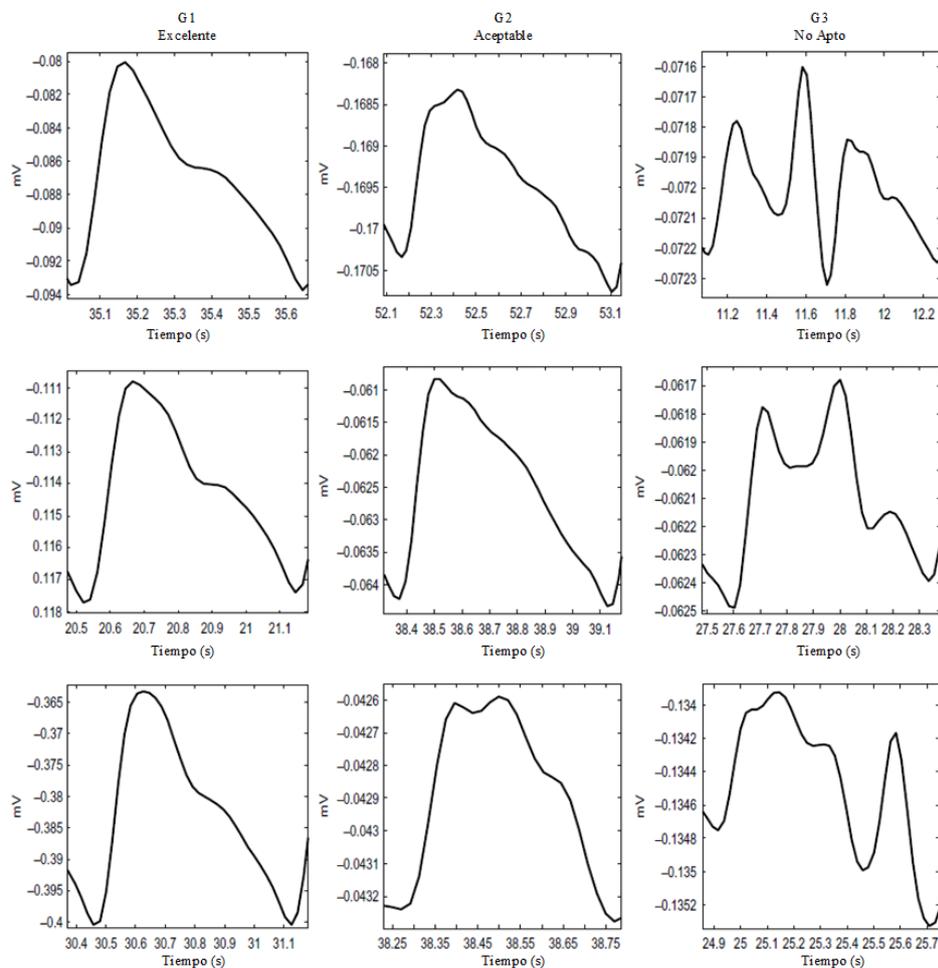
2.2.1.2.3. Calidad de señal PPG

Elgendi (2021) clasificó las señales PPG en 3 grupos en función de la calidad morfológica del ritmo más dominante dentro de la señal, como se muestra en la Figura 6:

- a. **Excelente para el diagnóstico:** Solo las señales PPG que tienen ondas sistólica y diastólica claramente definidas forman parte del grupo “excelente” para el diagnóstico (G1). Si hay más de dos ondas, significa que la forma de onda tiene ruido y, por lo tanto, puede pertenecer a uno de los otros dos tipos.
- b. **Aceptable para el diagnóstico:** Solo las señales PPG que tienen ondas sistólica y diastólica poco marcadas, pero que permiten calcular la FC, forman parte del grupo “aceptable” para el diagnóstico (G2).
- c. **No apto para el diagnóstico:** Solo las señales PPG que tienen mucho ruido y que no permiten calcular la FC ni diferenciar las ondas sistólica y diastólica forman parte del grupo “no apto” para el diagnóstico (G3).

Figura 6

Calidad de la Señal PPG



Nota. Cada columna corresponde a un grupo distinto, con tres ejemplos que ilustran la señal PPG completa (lado izquierdo). Fuente: Elgendi (2021).

2.2.1.3. Frecuencia respiratoria (FR)

La frecuencia respiratoria es la cantidad de inhalaciones y exhalaciones que están reguladas por el sistema nervioso y que realiza un ser vivo en un periodo determinado, generalmente se expresa en respiraciones por minuto (rpm). En la Tabla 1 se visualizan los valores normales de FR de acuerdo a la edad.

Tabla 1

Valores normales de FR en RPM

Edad	Frecuencia Respiratoria (RPM)
Neonato	30 – 50
0 – 5 meses	25 – 45
6 – 12 meses	20 – 40
1 – 3 años	20 – 35
3 – 5 años	20 – 30
6 – 12 años	15 – 30
13 – 18 años	14 – 20
Adultez	12 – 20
Vejez	12 – 16

2.2.1.3.1. Alteraciones en los valores normales de FR.

a. Taquipnea

Se denomina así a la frecuencia respiratoria cuando está por encima del límite superior de respiraciones por minuto sus causas más comunes son: ansiedad, fiebre, afecciones respiratorias, asma o neumonía, embolia pulmonar, etc. (Equimed, 2022).

b. Bradipnea

Se denomina así la frecuencia respiratoria cuando está por debajo del límite inferior de respiraciones por minuto, sus causas más comunes son: obstrucción de las vías respiratorias, sobredosis de drogas, apnea obstrusiva del sueño, traumatismo craneoencefálico (Equimed, 2022).

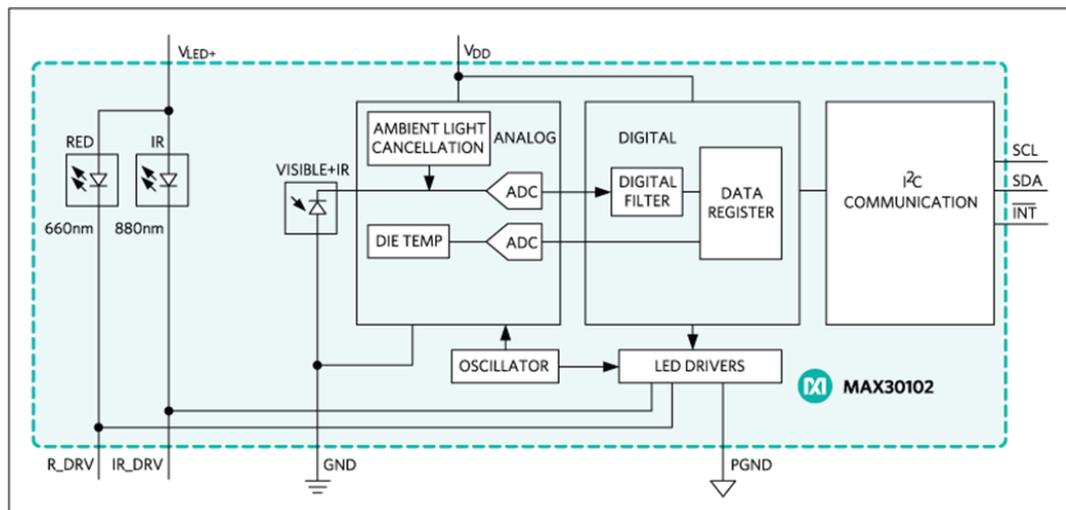
2.2.2. Sistemas electrónicos

2.2.2.1. Sensor digital MAX30102

Es un módulo integrado que combina un oxímetro de pulso y un monitor de ritmo cardíaco. Cuenta con dos luces LED: roja (660 nm) e infrarroja (880 nm), un fotodetector, óptica especializada, un filtro de luz ambiental de entre 50 y 60Hz, y un conversor ADC delta sigma de 18 bits y de hasta 1600 muestras por segundo. Además, tiene un sensor interno de temperatura para compensar los efectos de la temperatura en la medición (Hernández y Pimentel, 2023). En la Figura 7 se muestra su diagrama funcional.

Figura 7

Diagrama Funcional de MAX30102

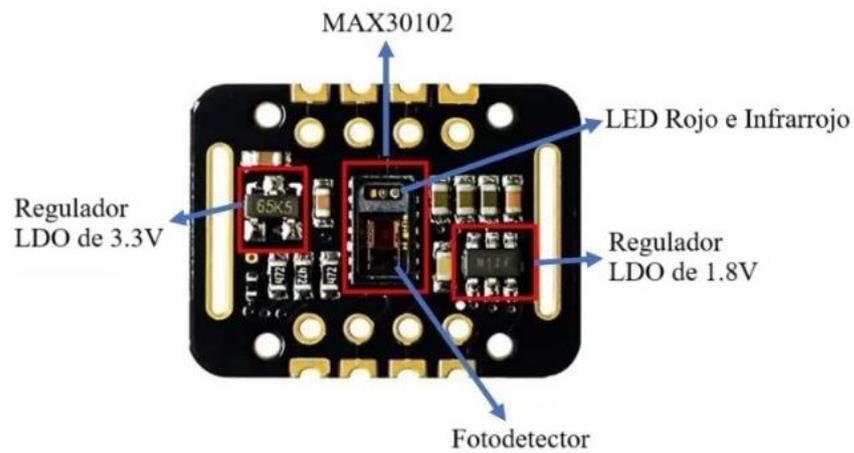


Nota. Tomado de Maxim Integrated (2018)

El chip requiere dos voltajes de suministro, por ello el módulo cuenta con dos reguladores de baja caída (LDO, por sus siglas en inglés), uno de 1.8V para el circuito integrado (CI) y otro de 3.3V para los LED, como se visualiza en la Figura 8, de modo que solo se necesita una fuente de 5V para alimentarlo. Su consumo de corriente es muy bajo, por lo cual es perfecto para aplicaciones portátiles. Puede ser empleado en dispositivos de monitoreo médico, asistentes de estado físico y wearables en general.

Figura 8

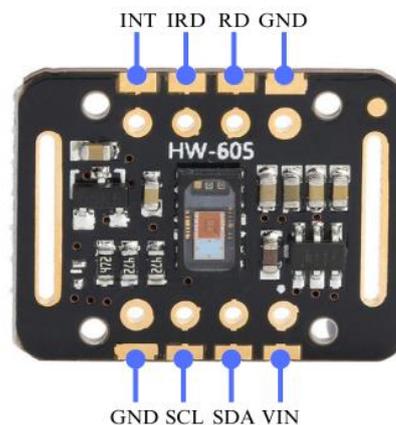
Módulo MAX30102



El módulo utiliza una interfaz I2C simple de dos cables para comunicarse con el microcontrolador. Tiene una dirección I2C fija: $0xAE_{HEX}$ (para operación de escritura) y $0xAF_{HEX}$ (para operación de lectura). El módulo MAX30102 consta de ocho pines como se observa en la Figura 9.

Figura 9

Pines del módulo MAX30102



En la Tabla 2 se describe la función de cada uno de los pines y en la Tabla 3 se describen las especificaciones técnicas del módulo MAX30102.

Tabla 2*Descripción de pines del módulo MAX30102*

Pin	Función
VIN	Este pin se utiliza para suministrar energía al sensor. Este sensor se enciende a 3.3 – 5.5 V.
SCL	Este es el pin del reloj serie I2C.
SDA	Este es el pin de datos serie I2C.
INT	Este es el pin de interrupción baja activa. La resistencia incorporada lo eleva a ALTO, pero cuando ocurre una interrupción, pasa a BAJO hasta que la interrupción se borra.
IRD	Punto de conexión del cátodo LED IR y del controlador LED.
RD	Punto de conexión del cátodo LED rojo y del controlador LED.
GND	Esto se utiliza para suministrar tierra a este sensor y está conectado al pin de tierra de la fuente.

Tabla 3*Especificaciones técnicas del módulo MAX30102*

Característica	Especificación
Fuente de alimentación	3.3 V a 5.5 V
Consumo de corriente	~600 μ A (durante las mediciones) ~0.7 μ A (durante el modo de espera)
Longitud de onda del LED rojo	660nm
Longitud de onda del LED infrarrojo	880nm
Rango de temperatura	-40 °C a +85 °C
Precisión de temperatura	\pm 1 °C

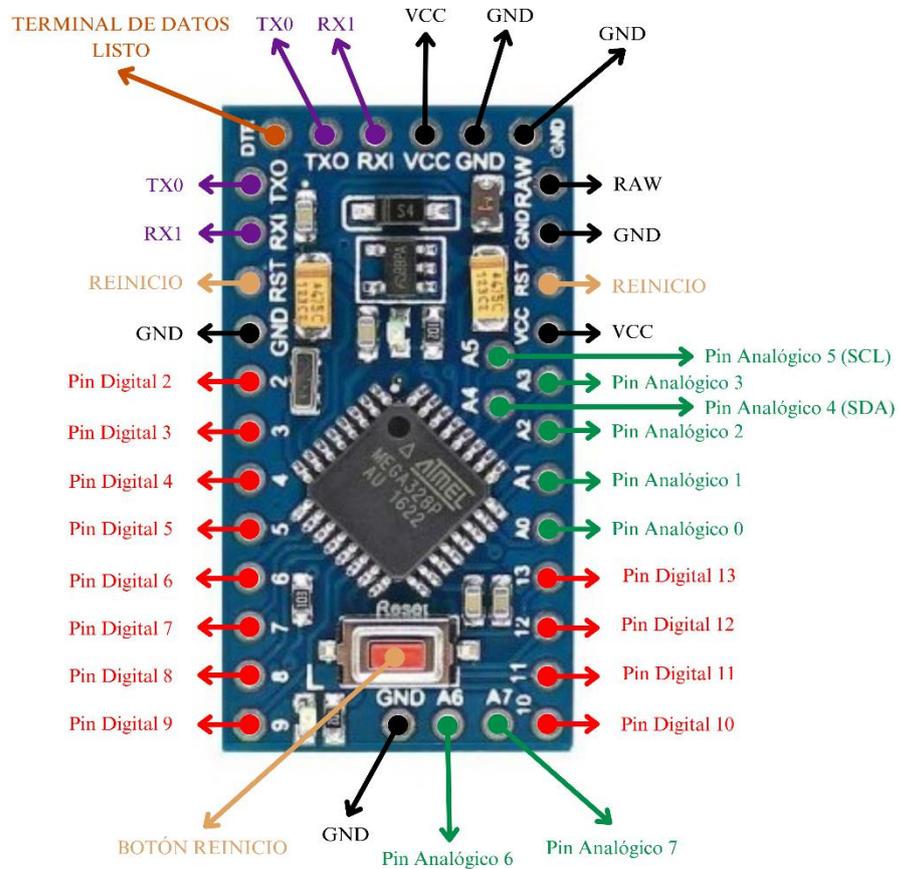
2.2.2.2. Arduino PRO Mini

Es una placa de desarrollo Arduino que cuenta con el microcontrolador ATmega328P, cuenta con 14 pines de entrada/salida digital, 8 entradas analógicas, y dependiendo del modelo, un cristal oscilador de 8 MHz y un regulador de voltaje

de 3.3V o un cristal oscilador de 16 MHz y un regulador de voltaje de 5V en placa, como se muestra en la Figura 10, ambos modelos cuentan con un botón de reinicio.

Figura 10

Pines Arduino Pro Mini



En la Tabla 4 se describe las funciones de algunos de sus pines.

Tabla 4

Pines con funciones especializadas

Pin	Función
RAW	Es un pin que se utiliza para suministrar tensión sin regular al módulo.
VCC	Fuente de alimentación regulada entre 3.3 a 5 voltios.
GND	Pines a tierra.

UART: 0 (RX) y 1 (TX)	Se usa para recibir (RX) y transmitir (TX) datos serie TTL.
PWM: 3, 5, 6, 9, 10 y 11	Proporciona salida PWM de 8 bits.
SPI: 10(SS), 11(MOSI), 12(MISO), 13(SCK)	Estos pines soportan la comunicación SPI, que, a pesar de que la proporciona el hardware subyacente, actualmente no está incluida en el lenguaje Arduino
LED: 13	Si el pin tiene un valor ALTO, el LED está encendido, y si el pin tiene un valor BAJO, está apagado.
I2C: A4 (SDA) y A5 (SCL)	Admite comunicación I2C mediante la biblioteca Wire.

En la Tabla 5 se muestran las especificaciones técnicas del microcontrolador ATmega328P.

Tabla 5

Especificaciones técnicas ATmega328P

Característica	Especificación
Fuente de alimentación de la placa	3.35 - 12V (modelo de 3.3V) o 5 - 12V (modelo de 5V)
Voltaje de funcionamiento del circuito	3.3 o 5V (según modelo)
Bus UART	1
Bus SPI	1
Bus I2C	1
Memoria Flash	32 KB, 2 KB para el arranque.
SRAM	2 KB
EEPROM	1 KB
Velocidad de Reloj	8 MHz (Versiones de 3.3V) o 16 MHz (Versiones de 5V)

2.2.2.3. Módulo Bluetooth HC-05

El módulo Bluetooth HC-05 es un dispositivo que posibilita comunicarse sin cables entre un Arduino y otros aparatos, como móviles, computadoras o módulos Bluetooth de otros Arduino. El módulo Bluetooth HC-05 se puede ajustar como maestro o esclavo, lo que le otorga más versatilidad para crear distintas clases de redes Bluetooth. El módulo Bluetooth HC-05 se relaciona con el Arduino por medio de un puerto serie, lo que simplifica el traspaso y la obtención de datos (Arduino, s. f.). El módulo Bluetooth HC-05 tiene seis pines como se muestra en la Figura 11.

Figura 11

Distribución de pines del módulo Bluetooth HC-05



En la Tabla 6 se describe la función de cada uno de los pines que compone el módulo Bluetooth HC-05.

Tabla 6

Función de pines del módulo Bluetooth HC-05

Pin	Función
EN	El estado del pin determina si el módulo funciona en modo de comando AT o en modo normal [Alto = modo de recepción de comandos AT (modo de respuesta de comandos), Bajo o NC = módulo Bluetooth funcionando normalmente]
VCC	Fuente de alimentación de 5V
GND	Pin a tierra
TXD	Transmite los datos en serie del módulo.
RXD	Recibe los datos en serie del módulo.
STATE	Pin que representa el estado del módulo.

El módulo Bluetooth HC-5 cuenta con las especificaciones técnicas mostradas en la Tabla 7 a continuación.

Tabla 7

Especificaciones técnicas del módulo Bluetooth HC-5

Característica	Especificación
Voltaje de operación	1.8 a 3.6V
Consumo de corriente	50mA
Estándar Bluetooth	Versión 2.0
Frecuencia	Banda ISM 2.4 GHz
Modulación	GFSK
Potencia de transmisión	4dBm
Sensibilidad	-80dBm
Alcance	10 metros
Interface de Comunicación	UART TTL
Velocidad de Transmisión	1200bps hasta 1.3Mbps

2.2.2.4. Batería Ion Litio

Las baterías recargables de ion de litio usan sustancias de litio como uno de sus electrodos. Estas baterías pueden guardar mucha energía en un espacio reducido, gracias a su alta densidad de energía. Además, tienen una tasa de autodescarga baja, lo que implica que no desperdician mucha energía cuando están en reposo. Poseen un cátodo y un ánodo. La Figura 12 muestra una batería de Ion Litio.

Figura 12

Batería Ion Litio



La Tabla 8 muestra las especificaciones técnicas de la batería a utilizar.

Tabla 8

Especificaciones técnicas de Batería Ion Litio

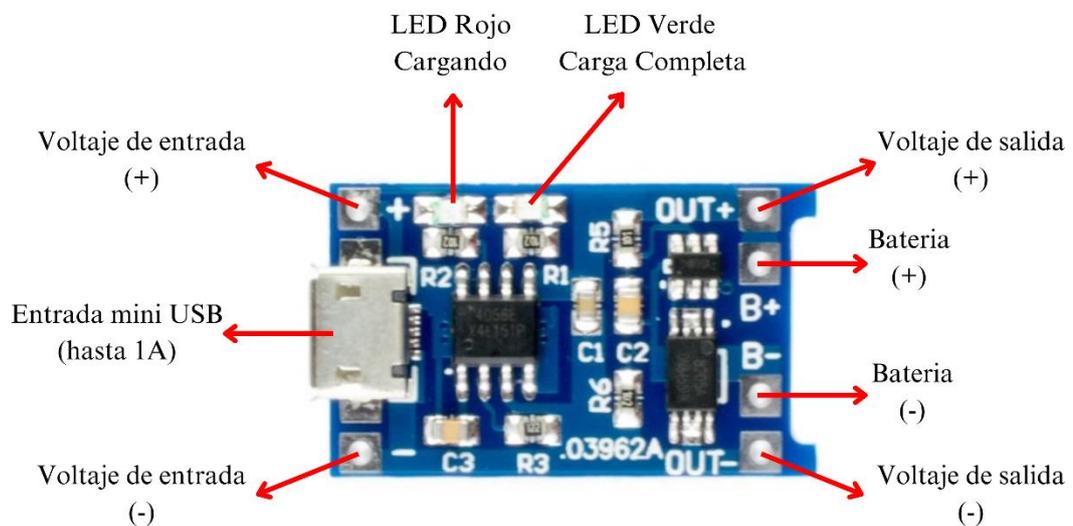
Característica	Especificación
Fabricante	ZIFCONN
Modelo	502040
Material	Polímero de litio
Voltaje nominal	3.7V DC
Voltaje de carga	4.2V
Capacidad	450mAh
Corriente máxima de carga	225mA (0.5C)
Corriente máxima de descarga	90mA (0.2C)

2.2.2.5. Módulo Cargador TP4056.

Es un módulo cargador USB de batería litio, con protección, permite cargar baterías de forma sencilla y segura. Funciona en baterías con voltajes de carga nominal de 3.7V y de carga completa a 4.2V. La corriente de carga es de 1A. En la Figura 13 se muestra el detalle de pines de conexión que lo componen.

Figura 13

Detalle de pines de conexión del módulo cargador TP4056



La Tabla 9 muestra las especificaciones técnicas del módulo cargador TP4056.

Tabla 9

Especificaciones técnicas del Cargador TP4056

Característica	Especificación
Voltaje de alimentación	4.5V a 5.5V DC
Cargador tipo	CC/CV (corriente constante/voltaje constante)
Corriente de carga en CC	1A
Voltaje de carga en CV	4.2V
Precisión de carga	1.5%
Tipo Entrada	Micro USB
Led Indicador Rojo	Cargando
Led Indicador Verde	Carga completa
Chip de carga	TP4056
Chip de protección	DW01G
Mosfet de carga/descarga	ML8205A

2.2.3. Procesamiento digital de señales

2.2.3.1. MATLAB

Es una plataforma de codificación y cálculo matemático empleada por numerosos ingenieros y científicos para el análisis de datos, el desarrollo de algoritmos y la creación de modelos. También posee una herramienta de diseño que permite crear apps profesionales, llamada App Designer.

2.2.3.2. Filtros digitales

Ballesteros y Torres (2018) mencionan que los filtros digitales son sistemas que transforman las señales en tiempo discreto para alterar su comportamiento en frecuencia. Estos sistemas se basan en el dominio Z o en una ecuación en diferencias para modelar las señales discretas. Si el sistema es lineal e invariante en el tiempo (LTI), el filtro digital se puede representar mediante la notación de ecuación en diferencias:

$$y[n] = - \sum_{k=1}^N a_k y[n - k] + \sum_{k=0}^M b_k x[n - k] \quad \dots (1)$$

Donde $y[n]$ es la señal filtrada (salida del sistema), $x[n]$ es la señal sin filtrar (entrada del sistema), a_k y b_k son los coeficientes de ponderación. El número de valores pasados de la señal filtrada que se usan para calcular el valor actual de la salida es N y el número de valores de la señal sin filtrar que se usan para calcular el valor actual de la salida es M . Cada una de estas señales, incluyendo el valor actual de la entrada, se multiplica por los coeficientes a_k y b_k , respectivamente.

Los filtros digitales se dividen en filtros de respuesta finita al impulso (FIR) y respuesta infinita al impulso (IIR).

2.2.3.2.1. Filtros de respuesta finita al impulso (FIR)

Según Escobar (2006), es aquel que tiene una respuesta finita al impulso y que se caracterizan por ser sistemas no recursivos. Partiendo de la ecuación (1) y haciendo $a_k = 0$, se obtiene la ecuación de diferencias de un filtro FIR.

$$y[n] = \sum_{k=0}^M b_k x[n - k] \quad \dots (2)$$

Un filtro FIR se puede expresar con la notación de la transformada Z. Partiendo de:

$$y[n] \xrightarrow{TZ} Y(z)$$

$$x[n - k] \xrightarrow{TZ} z^{-k} X(z)$$

Aplicando la Transformada Z a la ecuación (2) se obtiene la siguiente ecuación:

$$H(z) \equiv \frac{Y(z)}{X(z)} = \sum_{k=0}^M b_k z^{-k} \quad \dots (3)$$

Donde las constantes b_k corresponden a los coeficientes del filtro, así como también de la respuesta al impulso del sistema.

2.2.3.2.2. Diseño de filtros FIR

a. Método de las Ventanas.

Primero se selecciona la respuesta en frecuencia deseada $H_I(e^{j\omega})$, para ello se determina la respuesta al impulso $h_I(n)$. Como la $h_I(n)$ de la $H_I(e^{j\omega})$ es una respuesta infinita al impulso, es necesario truncarla en N puntos para generar un filtro practico. Se multiplica $h_I(n)$ por una ventana elegida de longitud N , con esto se logra una mejor respuesta espectral. La $h(n)$ obtenida es no causal (NC), ya que existen valores de $h(n)$ para $n < 0$, entonces es necesario desplazarla a la derecha, α puntos para obtener una versión causal. El efecto de recortar $h_I(n)$, ocasiona que su respuesta en frecuencia sea muy diferente a la ideal (Escobar, 2006).

La Tabla 10 muestra varias funciones de ventana cuyas respuestas en frecuencia son deseables.

Tabla 10

Funciones de ventana para filtros FIR

Nombre de la Ventana	Secuencia temporal, $h(n)$ $0 \leq n \leq M - 1$
Rectangular	$w(n) = \begin{cases} 1, & n = 0, 1, \dots, M - 1 \\ 0, & \text{en otro caso} \end{cases}$
Barlett (Triangular)	$1 - \frac{2 \left n - \frac{M-1}{2} \right }{M-1}$
Hanning	$\frac{1}{2} \left[1 - \cos \left(\frac{2\pi n}{M-1} \right) \right]$
Hamming	$0.54 - 0.46 \cos \left(\frac{2\pi n}{M-1} \right)$
Blackman	$0.42 - 0.5 \cos \left(\frac{2\pi n}{M-1} \right) + 0.08 \cos \left(\frac{4\pi n}{M-1} \right)$

Nota. Adaptado de Lizarzaburu y Villanueva (2023)

Estas funciones de ventana suavizan más la señal al aplicar la convolución en el dominio de la frecuencia y, por lo tanto, la zona de transición de la respuesta del filtro FIR se ensancha. Para disminuir el ancho de esta zona de transición, se puede incrementar la longitud de la ventana, lo que implica un filtro más grande (Lizarzaburu y Villanueva, 2023).

La Tabla 11 muestra estas características importantes en el dominio de la frecuencia de las diferentes funciones de ventana.

Tabla 11

Comparación de ventanas comúnmente utilizadas

Ventana	Amplitud de pico de los lóbulos laterales (relativa)	Anchura aproximada del lóbulo principal	Atenuación en la banda de rechazo	Ancho de banda de transición
Rectangular	-13 dB	$\frac{4\pi}{M+1}$	-21 dB	$\frac{1.81\pi}{M}$
Barlett	-25 dB	$\frac{8\pi}{M}$	-25 dB	$\frac{2.37\pi}{M}$
Hanning	-31 dB	$\frac{8\pi}{M}$	-44 dB	$\frac{6.22\pi}{\frac{M}{2}}$
Hamming	-41 dB	$\frac{8\pi}{M}$	-53 dB	$\frac{6.64\pi}{M}$
Blackman	-57 dB	$\frac{12\pi}{M}$	-74 dB	$\frac{11.12\pi}{M}$

Nota. Adaptado de Lizarzaburu y Villanueva (2023)

2.2.3.2.3. Filtros de respuesta infinita al impulso (IIR)

De la Rosa (s.f.), menciona que es aquel que tiene una respuesta al impulso que no se anula y que se distinguen por tener una realimentación de la señal de salida. En los filtros IIR, la salida depende no solo de la entrada actual y de las anteriores, sino también de las salidas previas. Es decir, son filtros recursivos (tienen realimentación), y por eso se espera que (en general) tengan una respuesta impulsional infinita. En el caso de los Filtros IIR, las constantes a_k son diferentes de cero y la ecuación de diferencias del filtro, se expresa como:

$$y[n] = -a_1y[n-1] - a_2y[n-2] - \dots - a_Ny[n-N] + b_0x[n] + b_1x[n-1] + b_2x[n-2] + \dots + b_Mx[n-M] \quad \dots (4)$$

La ecuación (5) representa la forma general de un filtro de respuesta infinita al impulso, donde los coeficientes a_k y b_k son distintos de cero, con M como el retardo máximo de la entrada y N como el retardo máximo de la salida (Ballesteros y Torres, 2018).

De forma similar al apartado anterior, se calcula la Transformada Z a la ecuación en diferencias. En este caso, Y(z) se define como:

$$H(z) \equiv \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad \dots (5)$$

2.2.3.2.4. Diseño de filtros IIR

a. Aproximación Butterworth

El filtro Butterworth es el mejor término medio entre atenuación y la respuesta en fase, también llamado filtro de Banda de paso máximamente plana. El filtro Butterworth es una de las aproximaciones más usadas (UDLA, s.f.).

Según Pactitis (2007), la ecuación general para una respuesta de amplitud del filtro Butterworth está definida de la siguiente manera:

$$|H(\omega)| = \frac{1}{\sqrt{1 + \left(\frac{\omega}{\omega_c}\right)^{2n}}} \quad \dots (6)$$

$$|H(0)| = 1 \quad \dots (7)$$

$$|H(\omega=\omega_c)| = \frac{1}{\sqrt{2}} \quad \dots (8)$$

$$|N(j\omega)|^2 = \frac{H^2}{1 + a_N \omega^{2n}} \quad \dots (9)$$

Por lo que el coeficiente a_N por lo general se le denota por ε^2 :

$$|N(j\omega)|^2 = \frac{H^2}{1 + \varepsilon^2 \omega^{2n}} \quad \dots (10)$$

Para determinar el valor de ε , se parte de la definición de atenuación máxima A_{max} :

$$A_{max} = 20 \log|N_{(j0)}| - 20 \log|N_{(j1)}| = 20 \log \sqrt{1 + \varepsilon^2} \quad \dots (11)$$

Despejando ε , se tiene:

$$\varepsilon = \sqrt{10^{\frac{A_{max}}{10}} - 1} \quad \dots (12)$$

Mediante la siguiente fórmula se puede determinar el orden del filtro, empezando por A_{max} que está dado por:

$$A_{max} = 20 \log|N_{(j\omega)}| - 20 \log|N_{(j\omega_s)}| = 20 \log \sqrt{1 + \varepsilon^2 \omega_s^{2n}} \quad \dots (13)$$

Despejando n , se obtiene:

$$n \geq \log \left[\frac{10^{0.1A_{min}} - 1}{\log \left(\frac{\omega_s}{\omega_c} \right)} \right]^{\frac{1}{2}} \quad \dots (14)$$

Su función de transferencia sería:

Para “ n ” par.

$$H(s) = \prod_{k=1}^{\frac{n}{2}} \frac{1}{s^2 + 2s \operatorname{sen} \left(\frac{(2k-1)\pi}{2n} \right) s + 1} \quad \dots (15)$$

Para “ n ” impar.

$$H(s) = \frac{1}{(s+1)} = \prod_{k=1}^{\frac{n-1}{2}} \frac{1}{s^2 + 2s \operatorname{sen} \left(\frac{(2k-1)\pi}{2n} \right) s + 1} \quad \dots (16)$$

III. METODOLOGÍA

3.1. Descripción de la metodología

La presente investigación es de naturaleza aplicada, experimental, con una orientación cuantitativa, ya que se basa en datos que se pueden medir y cuantificar, a los que se accede mediante observaciones y mediciones.

3.2. Implementación de la investigación

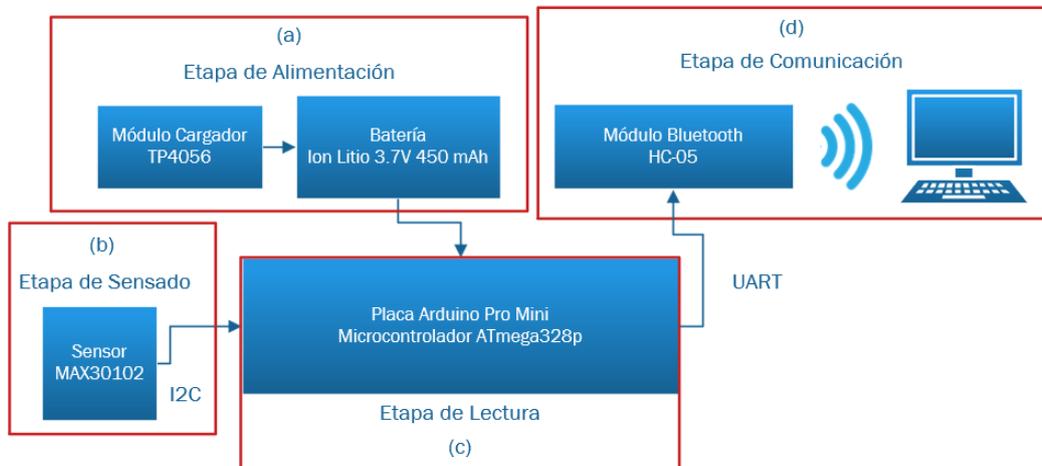
3.2.1. Desarrollo de Hardware

3.2.1.1. Descripción de diseño y construcción.

El sistema constara de 4 etapas: Etapa de alimentación, Etapa de sensado, Etapa de lectura y Etapa de comunicación, como se muestra en la figura 14.

Figura 14

Diagrama de bloques del hardware



Para la etapa de alimentación (a), se utilizó una batería Ion Litio de 3.7V y 450 mAh, que ira conectada a la placa Arduino, al ser este un equipo portable se utilizó también el micro cargador TP4056 que es un módulo cargador USB con protección, que permite cargar baterías de forma sencilla y segura.

La etapa de sensado (b) se compone fundamentalmente por el sensor MAX30102, este dispositivo consiste en dos LED, uno rojo y otro infrarrojo, que emiten luz sobre la piel a diferentes longitudes de onda (660nm y 880nm respectivamente), y un fotodiodo que mide la intensidad de luz reflejada debido a la variación del volumen sanguíneo de los tejidos y la envía como señal eléctrica al

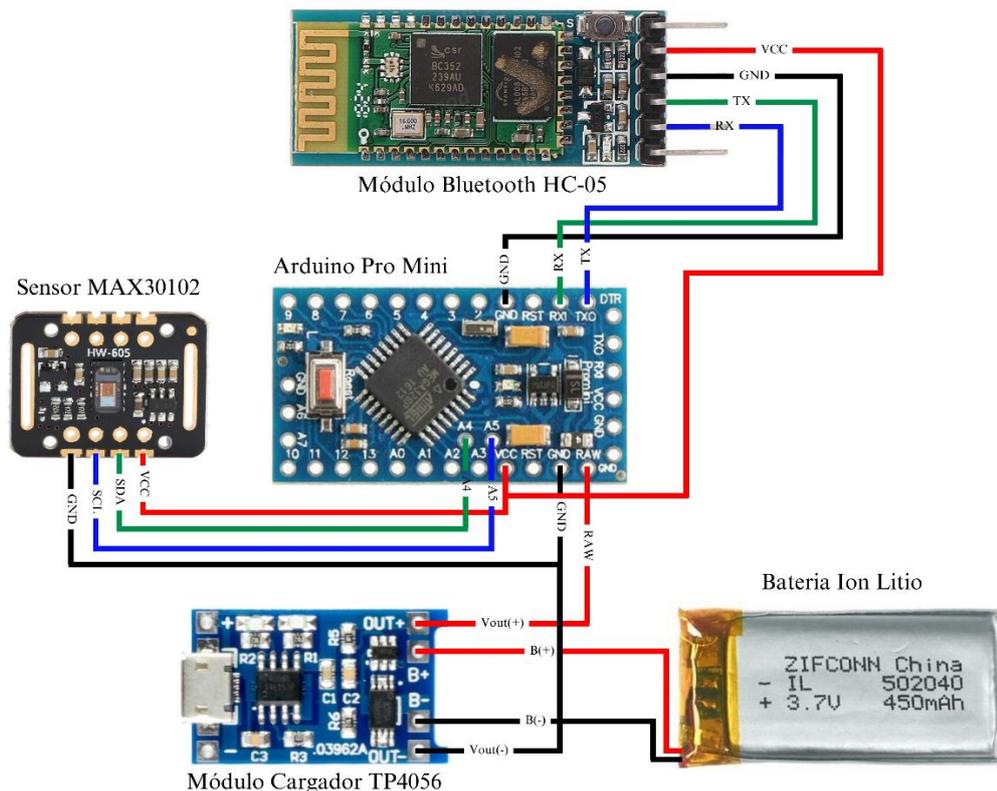
circuito integrado del sensor, que la amplifica, la filtra y la convierte en un valor digital. Este valor digital representa la señal PPG.

Para la etapa de lectura (c), se utilizó la tarjeta de desarrollo Arduino Pro Mini, que posee el microcontrolador ATmega328p, en el que se carga el programa o sketch que permitirá leer la data sin procesar capturada por el sensor mediante comunicación I2C, para cargar dicho programa a la tarjeta Arduino se utilizó el conversor USB-Serial TTL FT232RL, que se conecta a través del puerto UART TTL del Arduino.

Por último, para la etapa de comunicación (d), se utilizó el módulo Bluetooth HC-05, el cual se conecta mediante comunicación UART TTL con el Arduino y que permitirá transmitir los datos captados en la etapa de lectura hacia una PC de manera inalámbrica, para esto se debe configurar el módulo en modo esclavo, de esta manera los datos que ingresen por el pin RX del módulo serán transmitidos hacia la PC en la cual se completará el procesamiento de los datos. En la Figura 15 se aprecia la conexión entre los 4 módulos que componen la construcción del hardware.

Figura 15

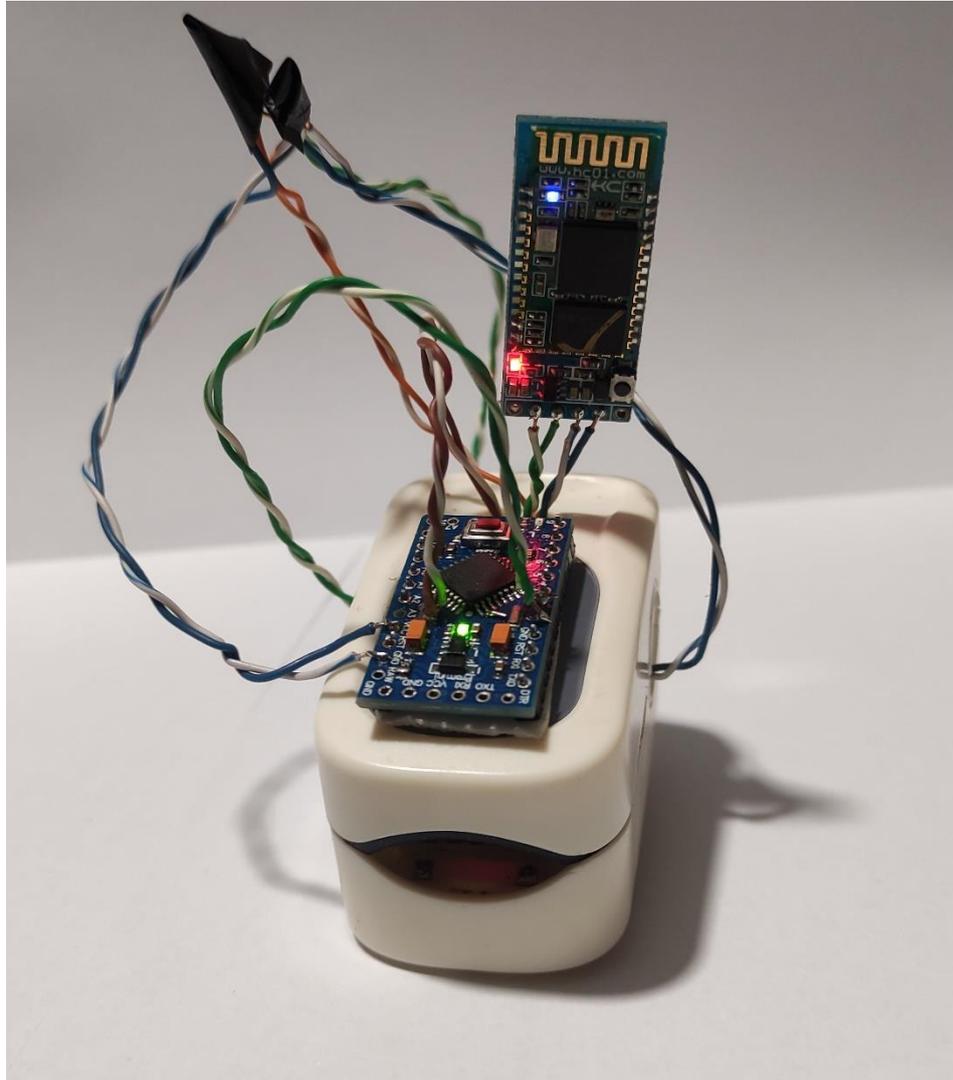
Diagrama de conexión del dispositivo



Las conexiones fueron realizadas según la Figura 15, utilizando los buses I2C (Pines A4 y A5) y el Puerto Serial UART TTL (Pines RX1 y TX0) del Arduino Pro Mini. El prototipo del dispositivo totalmente conectado se puede apreciar en la figura 16.

Figura 16

Prototipo final

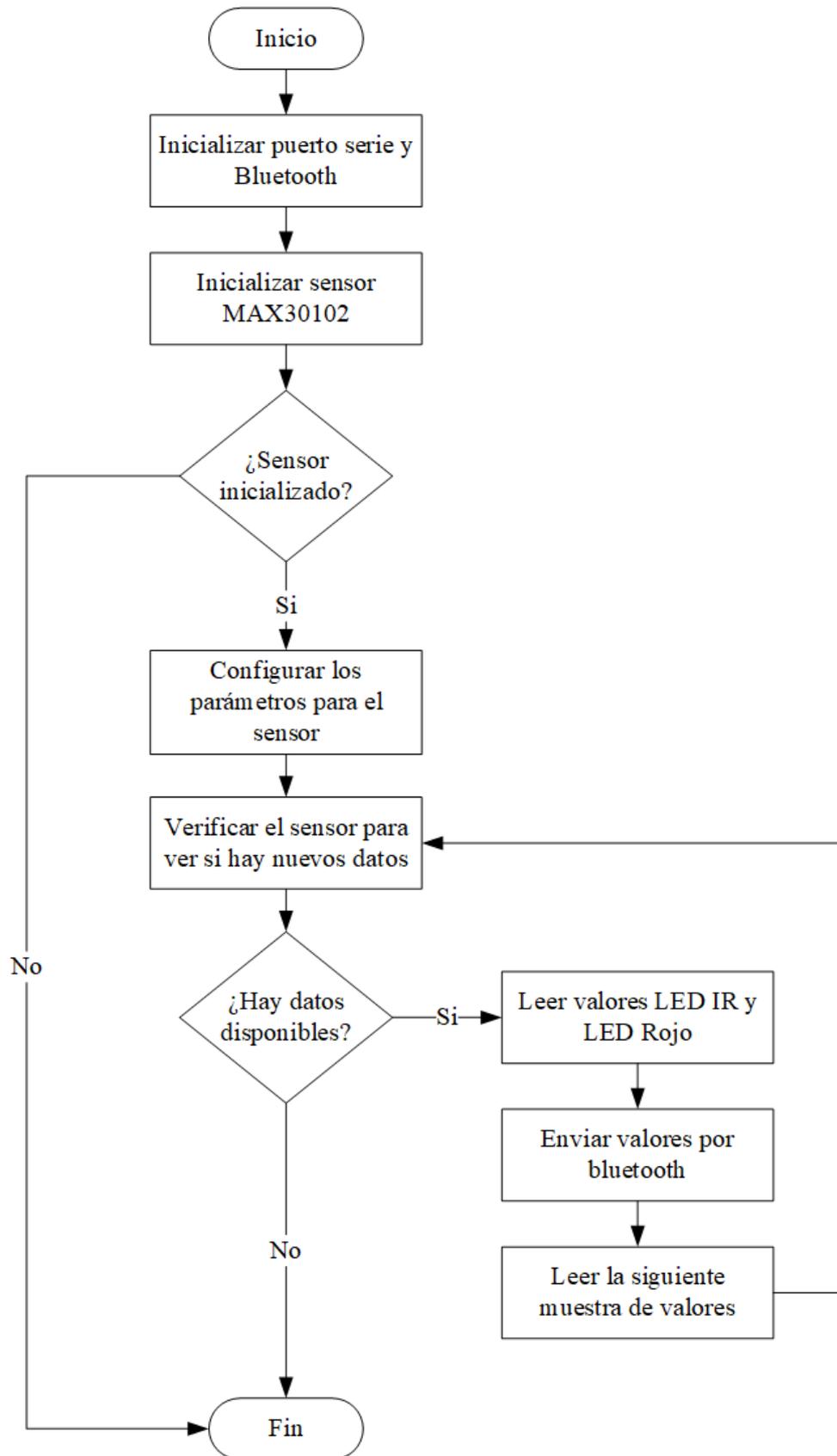


3.2.1.2. Adquisición y representación de datos.

La adquisición de datos consiste en la obtención de los valores ADC que representan las señales del LED Rojo y LED Infrarrojo. Para ello se utilizó el código descrito en el Anexo 3, del cual se muestra el diagrama de flujo en la Figura 17 a continuación.

Figura 17

Diagrama de flujo del código en Arduino



Con el hardware conectado a la PC se carga y compila el código mencionado anteriormente, terminado el proceso, se abre el monitor serie del IDE de Arduino y se establece la velocidad en baudios en 115200 y se procede a colocar el dedo sobre el sensor. El sensor MAX30102 posee un conversor ADC de 18 bits de resolución como se indicó en el marco teórico, por lo tanto, la señal digitalizada trabaja en un rango de 0 a 262143 ($2^{18} = 262144$). En la Tabla 12 se observa que los valores crudos impresos se encuentran en ese rango.

Tabla 12

Valores ADC del LED Infrarrojo y LED Rojo

Valores ADC LED Infrarrojo	Valores ADC LED Rojo
215458	253332
215455	253354
215449	253356
215495	253382
215463	253303
215464	253361
215456	253370
215470	253357
215487	253391
215471	253366
215459	253357
215464	253342
215447	253332
215440	253356
215469	253349
215453	253329
215461	253330
215460	253304
215423	253338
215457	253352
215446	253361

Nota. Muestra de 20 primeros datos ADC del sensor MAX30102.

Se ha desarrollado un código en MATLAB para recibir los datos que se transmiten por medio de comunicación bluetooth desde la placa de Arduino hacia el script creado en MATLAB (ver Anexo 4).

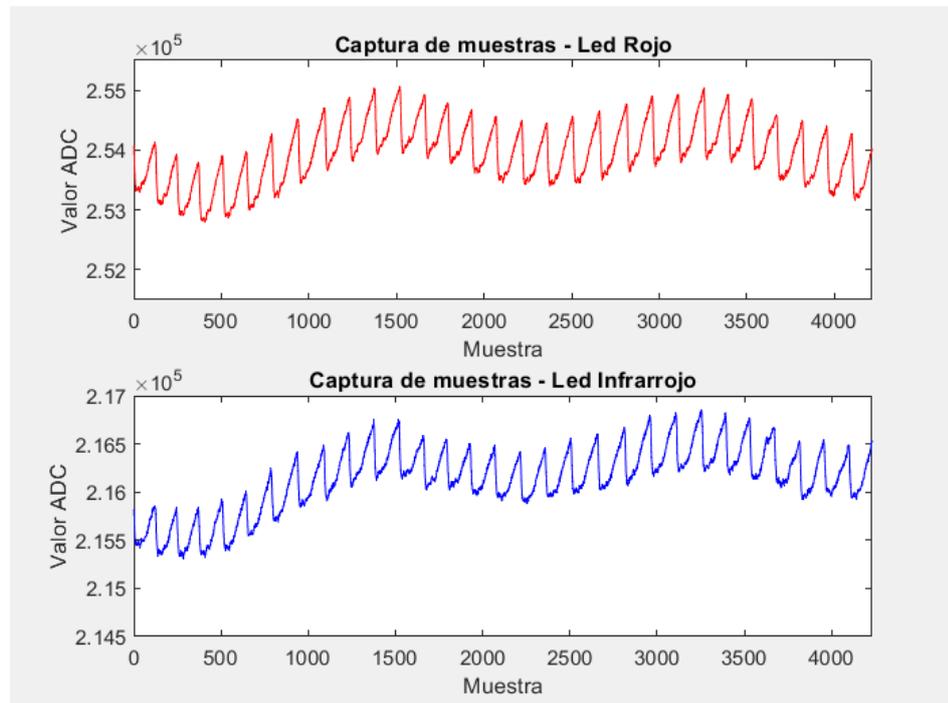
Primeramente, se define el puerto serial y la velocidad de comunicación que en este caso será de 115200 baudios, así como la cantidad de muestras que se desea adquirir, en este caso se establecen en 38 400 muestras a una frecuencia de muestreo

de 400Hz que equivalen a 4 min, aproximadamente. Luego de ello se crean dos vectores en los cuales se almacenarán los datos obtenidos de ambos LED del sensor. Al terminar la lectura se cierra el puerto de comunicación y se crea el archivo “.mat” cuyo nombre estará definido por la fecha y hora en la que se realizó la lectura, en este se guardará los datos para su posterior análisis. Finalmente se grafican los datos obtenidos por el sensor.

Se seleccionaron las señales más apropiadas, lo cual se basó en la comparación gráfica que usa Elgendi (2021) de la Figura 6, de esa forma se logra conservar para nuestro estudio, aquellas señales de las se puede extraer las características necesarias. La figura 18 representa la forma de una señal apta para ser procesada.

Figura 18

Gráfica de los datos ADC recibidos en Matlab



Nota. En la figura se muestra la representación gráfica de los valores ADC obtenidos por el fotodetector del sensor con respecto a ambos LED, rojo (superior) e infrarrojo (inferior).

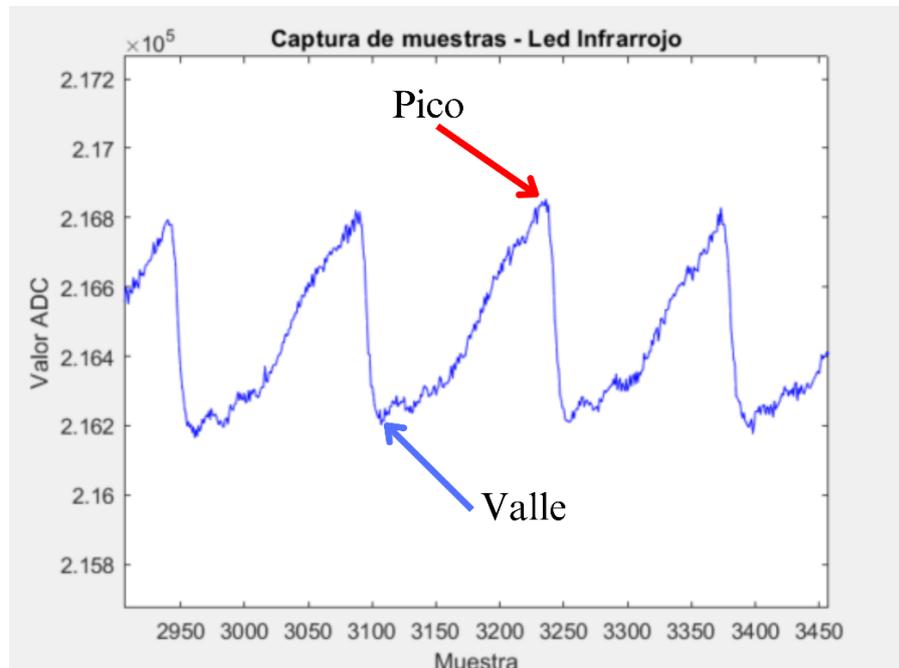
De las dos señales que usa el MAX30102, la señal infrarroja (880 nm) se relaciona más con la frecuencia respiratoria, ya que se refleja más en la sangre

oxigenada, que es la que circula por las arterias. De acuerdo a esto y teniendo en cuenta que solo se requiere uno de los LED para generar una señal PPG, a partir de este punto se ha trabajado únicamente con la señal generada por el LED Infrarrojo.

En la Figura 19, los puntos más altos y más bajos de la amplitud de la señal, se denominan picos y valles, respectivamente. los picos representan los momentos en los que el flujo sanguíneo es máximo, debido a la contracción del corazón, y los valles representan los momentos en los que el flujo sanguíneo es mínimo, debido a la relajación del corazón.

Figura 19

Acercamiento de la señal PPG



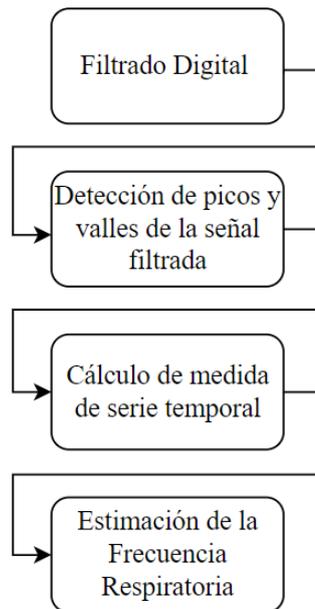
Nota. La figura muestra un acercamiento de la señal infrarroja de la Figura 18, en ella se señala el valle y el pico para un periodo de la señal.

3.2.2. Algoritmo Propuesto

Para estimar la frecuencia respiratoria es necesario hacer un procesamiento adecuado de la señal PPG. La Figura 20 muestra los pasos a seguir para lograr el objetivo planteado: Filtrado digital de la señal, detección de picos de la señal filtrada, cálculo de la medida de serie temporal y, por último, la estimación de la frecuencia respiratoria en RPM.

Figura 20

Pasos del algoritmo propuesto

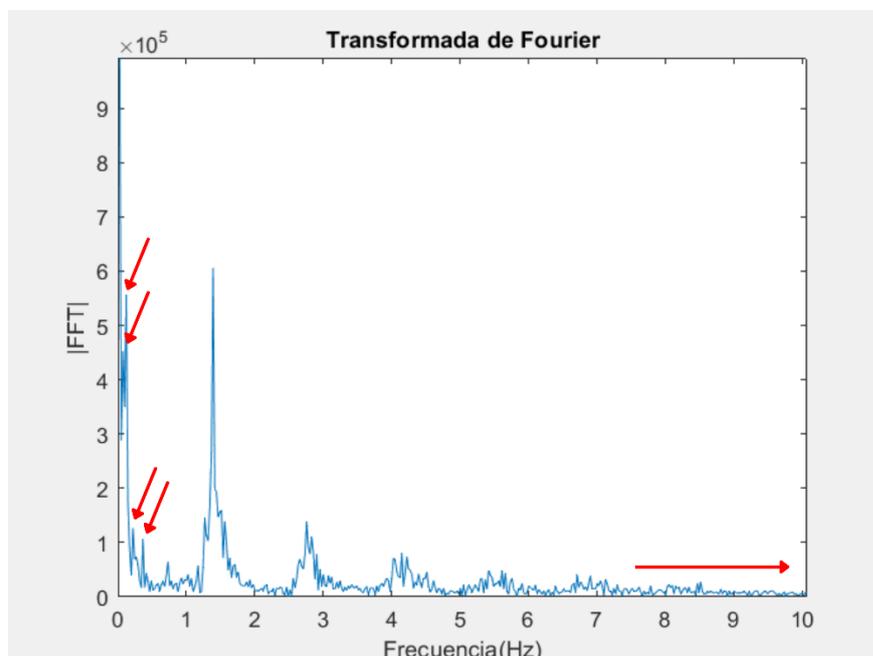


3.2.2.1. Filtrado Digital de la señal PPG

A la señal del LED infrarrojo del sensor graficada en la Figura 18 se le aplicó la Transformada rápida de Fourier (FFT), para estudiar la respuesta en frecuencia, como se muestra en las Figura 21.

Figura 21

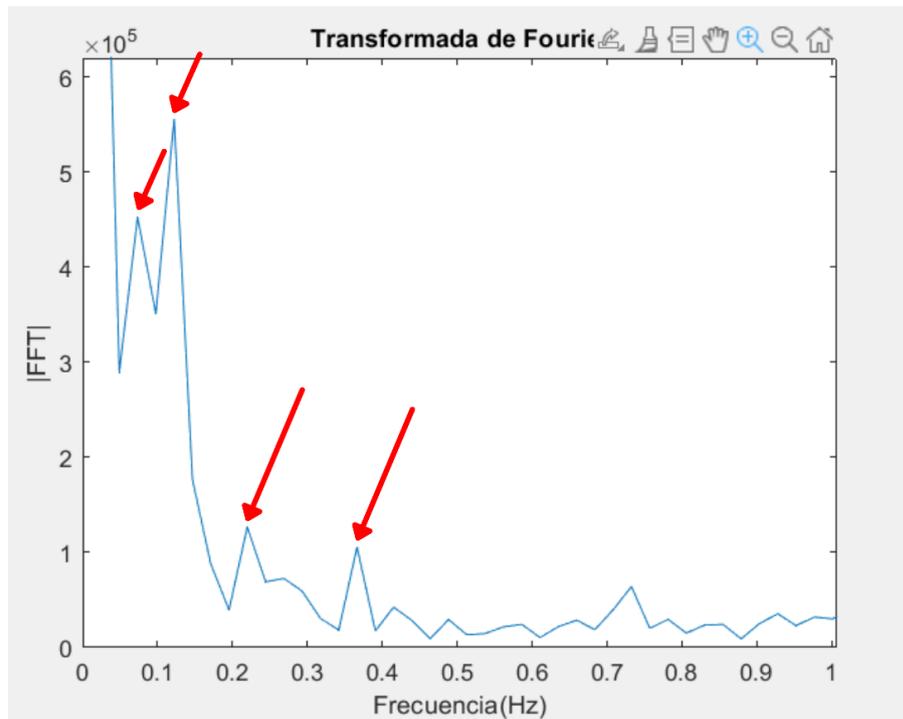
Transformada rápida de Fourier de la señal infrarroja



En la Figura 21 se observa que la componente principal de la FFT se encuentra entre 0.2 Hz y 1 Hz aproximadamente y que la fracción de la señal que brinda información relevante acerca de la señal original, por lo tanto, de la frecuencia respiratoria, se ubica hasta 9 Hz aproximadamente, a partir de ese punto ya no se distingue mucho una componente de la señal original, esta puede ser confundida con ruido. Ahora bien, haciendo un acercamiento a la figura anterior se obtiene la Figura 22.

Figura 22

Acercamiento de la Transformada rápida de Fourier de la señal infrarroja



En la Figura 22, se observan 5 componentes de gran amplitud antes de 0.4 Hz, siendo la primera componente la más representativa e indeseada, el ruido oscilatorio de la señal, es decir, la componente DC.

Teniendo en cuenta el análisis de ambas graficas se procedió con el diseño de un filtro de aproximación Butterworth de paso de banda en MATLAB (ver Anexo 5) se utilizó un orden 3, ya que un orden mayor generaba inestabilidad en la señal, la frecuencia de muestreo fue de 200 Hz, que es la mitad de la frecuencia de muestreo del sensor, y las frecuencias de corte fueron, en el caso de la señal de la

gráfica, 0.2 Hz y 0.33 Hz para permitir solo las frecuencias dentro de ese rango de frecuencias (que corresponden al rango de 12 – 20 rpm).

La función de transferencia del filtro está definida de la siguiente manera:

$$H(z) = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + b_3z^{-3}}{1 + a_1z^{-1} + a_2z^{-2} + a_3z^{-3}} \quad \dots (17)$$

Donde los coeficientes b_i y a_i dependen de los parámetros del filtro analógico y del periodo de muestreo. La ecuación de diferencias que corresponde a esta función de transferencia es:

$$y[n] = -a_1y[n-1] - a_2y[n-2] - a_3y[n-3] + b_0x[n] + b_1x[n-1] + b_2x[n-2] + b_3x[n-3] \quad \dots (18)$$

Donde $x[n]$ es la entrada y $y[n]$ es la salida del filtro.

Para el diseño de nuestro filtro en el software de MATLAB se utiliza la siguiente instrucción:

$$[B, A] = \text{butter}(N, fn, type)$$

Donde:

- B : hace referencia a los coeficientes del numerador de $H(z)$.
- A : hace referencia a los coeficientes del denominador de $H(z)$.
- N : es el orden del filtro digital.
- fn : es la frecuencia de corte normalizada del filtro digital.
- $type$: se refiere al tipo de filtro a diseñar, puede ser:
 - filtro pasa altos: high
 - filtro pasa bajos: low
 - filtro pasa banda: bandpass
 - filtro rechaza-banda: stop

En la figura 23 se observa un fragmento del código en MATLAB para el diseño del filtro y su respuesta en frecuencia.

Figura 23

Código de diseño de filtro en MATLAB

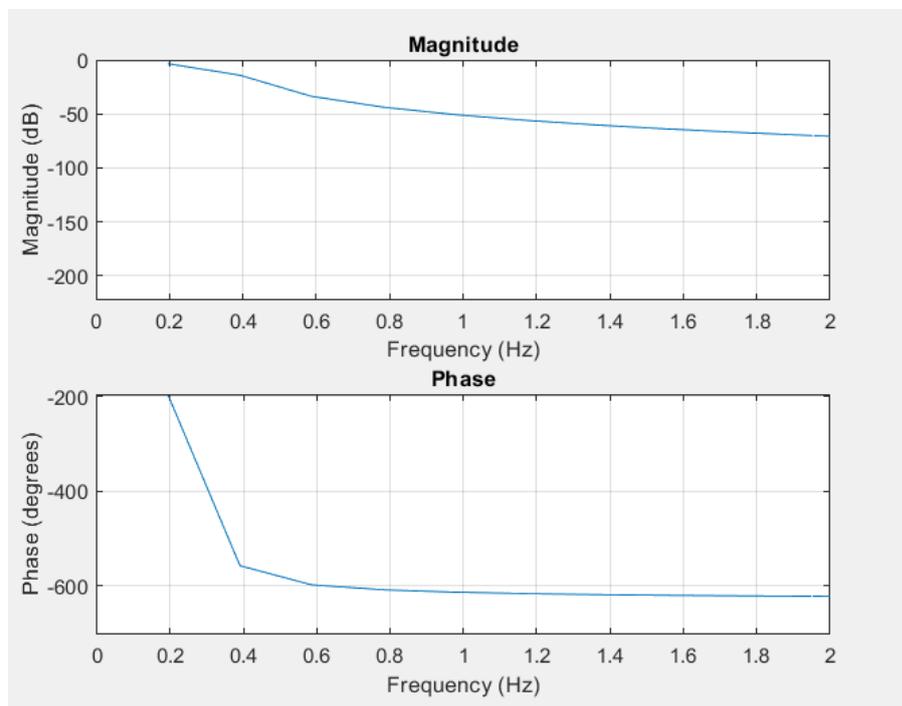
```
% Filtro
Fs=200; % Frec. de muestreo del sensor
n=3; % Orden del filtro
wn = [0.2 0.33]/(Fs/2); % Puntos de corte
[num,den]=butter(n,wn,'Bandpass');

freqz(num,den,[],Fs); % Respuesta del filtro en frec.
```

En la Figura 24 se observa la respuesta en magnitud y en fase del filtro Butterworth pasa banda.

Figura 24

Gráfica de respuesta del filtro en MATLAB

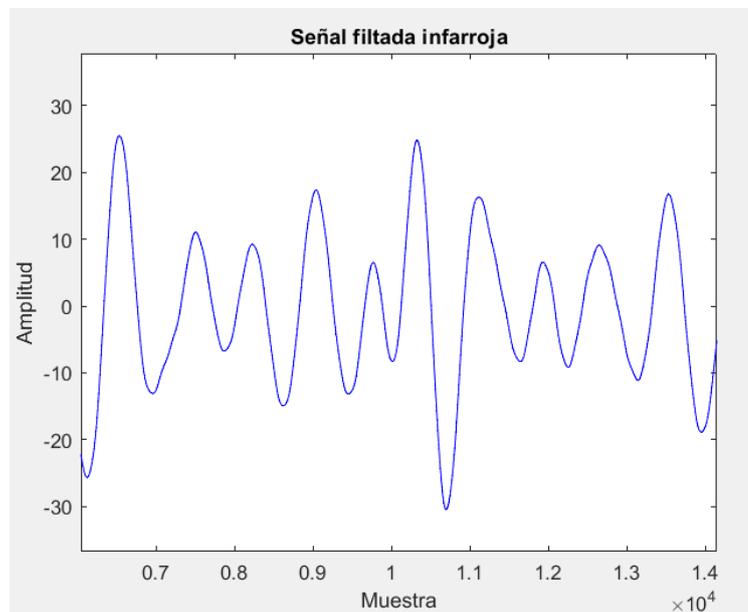


Nota. La gráfica de magnitud (parte superior), muestra la atenuación de la señal para los valores que no se encuentran en el intervalo de frecuencias de corte en el filtro, mientras que la gráfica de fase (parte inferior), indica un desfase negativo, quiere decir que la señal se retrasa.

Por último, se aplica el filtro a la señal del LED infrarrojo que se está analizando y se obtiene la gráfica que se muestra en la Figura 25. La Figura 26 corresponde a la señal filtrada en el dominio de la frecuencia.

Figura 25

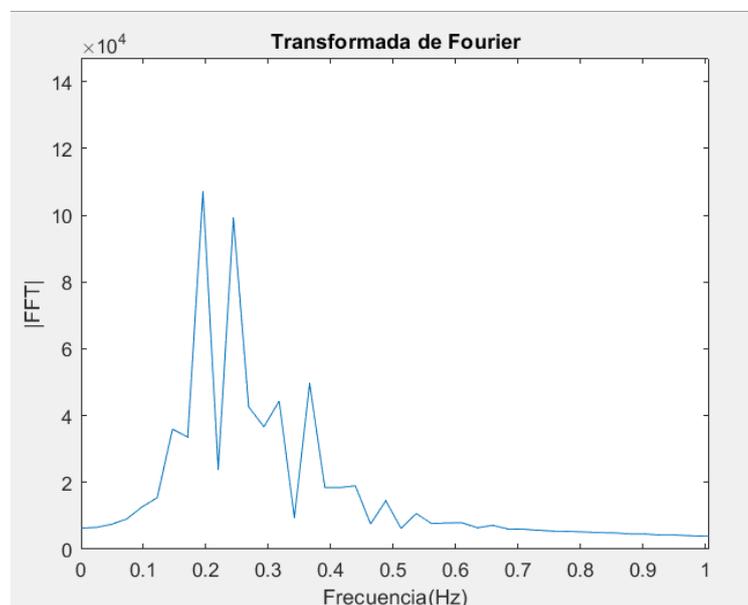
Señal PPG del LED infrarrojo filtrada



Nota. Señal PPG del LED infrarrojo, filtrada en los rangos de frecuencias de 0.2 a 0.33 Hz.

Figura 26

Transformada de Fourier de la señal PPG filtrada del LED infrarrojo



3.2.2.2. Detección de picos y valles en la señal filtrada

Para la detección de picos en el dominio del tiempo de la señal filtrada se implementó un script en MATLAB (ver el anexo 6), en el cual se utilizó la función 'findpeaks', esta función permite especificar varios parámetros para ajustar el proceso de detección de picos. En este código se utiliza para identificar tanto los picos como los valles de la señal PPG filtrada. Los parámetros utilizados son: 'MinPeakProminence', que se refiere al umbral de prominencia mínimo que un pico o valle debe tener para ser considerado y 'MinPeakDistance' que hace referencia a la distancia mínima entre picos o valles, como se puede observar en el fragmento de código de la Figura 27.

Figura 27

Parámetros de la función 'findpeaks'

```
% Configurar los parámetros para la función findpeaks
up = 2; % Umbral de prominencia para identificar picos y valles
dm_p = 100; % Distancia mínima entre picos y valles

% Encontrar los picos en la señal PPG
[p, ubi_p] = findpeaks(signal_PPG, ...
    'MinPeakProminence', up, 'MinPeakDistance', dm_p);

% Invertir la señal para encontrar los valles
signal_PPG_invertida = -signal_PPG;

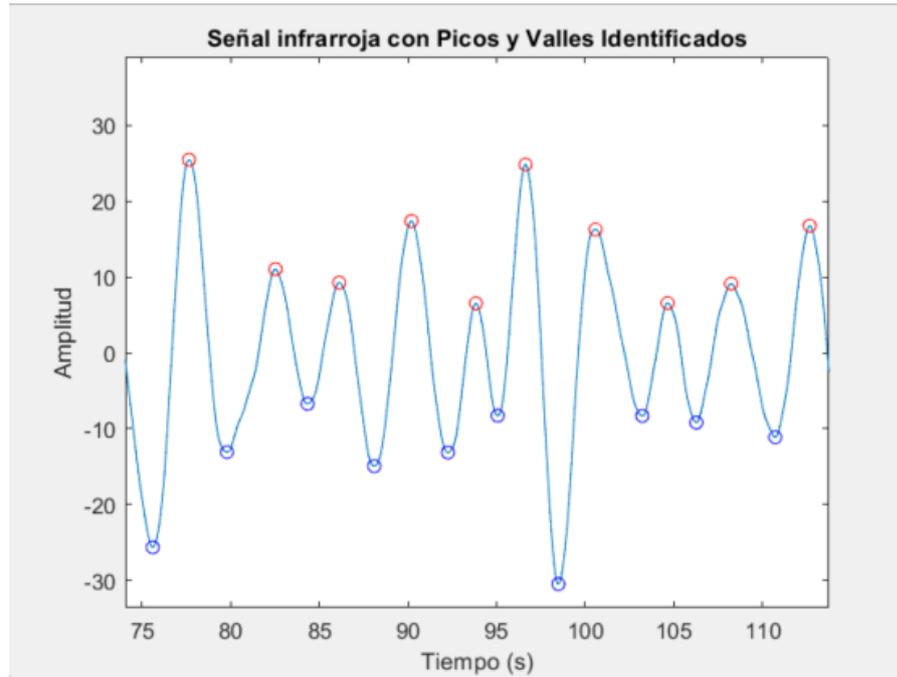
% Encontrar los valles en la señal PPG invertida
[v, ubi_v] = findpeaks(signal_PPG_invertida, ...
    'MinPeakProminence', up, 'MinPeakDistance', dm_p);
```

Nota. Fracción de código para encontrar los picos y valles de la señal filtrada del LED infrarrojo.

En la figura 28, se muestran marcados con un círculo rojo y azul los puntos correspondientes a los picos y valles de la señal filtrada, respectivamente, que a su vez han sido almacenados en vectores para su fácil manejo en MATLAB.

Figura 28

Picos y valles de la señal filtrada



3.2.2.3. Cálculo de medida de serie temporales.

Las series temporales de picos y valles se pueden utilizar para derivar tres nuevas series temporales que representan diferentes fuentes de información relacionada con la respiración. Una de ellas es la variación de frecuencia inducida por la respiración (RIFV, por sus siglas en inglés), esta representa el cambio en el valor de la frecuencia cardíaca instantánea (FC) durante el ciclo respiratorio, el RIFV se determina mediante el intervalo de tiempo entre picos o valles sucesivos. El cálculo de estos intervalos se puede expresar de la siguiente manera:

Si se tiene un vector de tiempos de picos t_{picos} , donde cada elemento $t_{picos}[i]$ representa el tiempo en el que ocurre el i –ésimo pico, entonces el intervalo de tiempo entre picos ($PP_{intervalos}$) se puede calcular como la diferencia entre tiempos consecutivos.

$$PP_{intervalos}[i] = t_{picos}[i + 1] - t_{picos}[i] \quad \dots (19)$$

Aquí, i varía desde 1 hasta $N - 1$, donde N es la cantidad total de picos detectados. El resultado es un vector de intervalos de tiempo entre picos.

También se puede expresar como una sumatoria:

$$PP_{intervalos}[i] = \sum_{k=1}^i (t_{picos}[k+1] - t_{picos}[k]) \quad \dots (20)$$

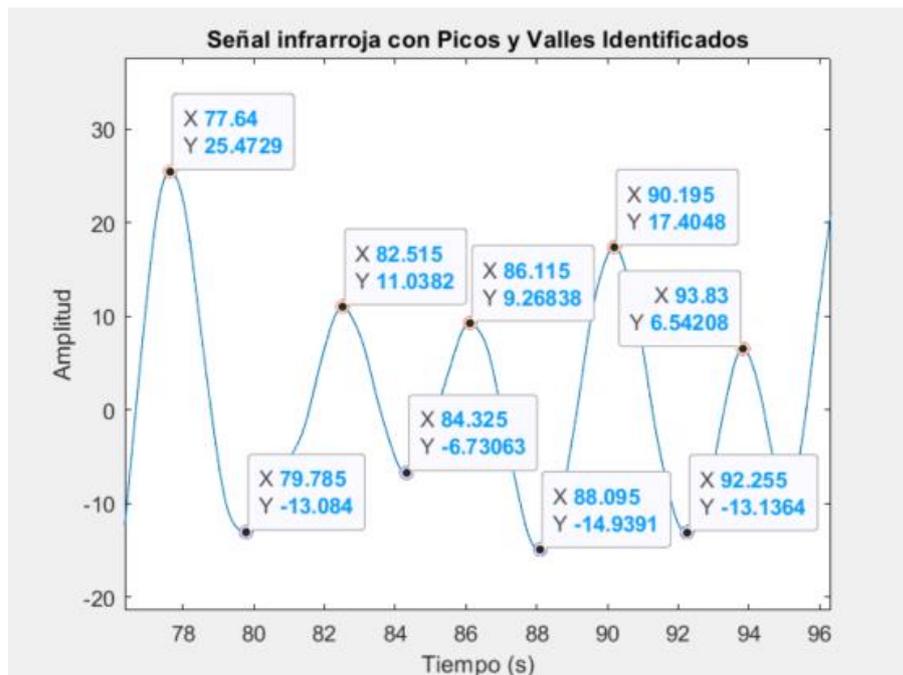
Donde la sumatoria se realiza hasta el i –ésimo pico, obteniendo, así como resultado el intervalo total acumulado hasta ese punto. De igual manera se procede con los valles de la señal.

$$VV_{intervalos}[i] = \sum_{k=1}^i (t_{valles}[k+1] - t_{valles}[k]) \quad \dots (21)$$

Donde $VV_{intervalos}$ es el intervalo de tiempo entre valles y t_{valles} es el vector de tiempo entre valles. En la Figura 29 se ilustra un ejemplo.

Figura 29

Tiempo correspondiente a las ubicaciones de los picos y valles



Entonces, tomando de referencia la señal infrarroja de la Figura 29 se ubica el primer pico a 77.64s, así mismo el segundo pico se ubica a 82.51s, esto quiere decir que el primer intervalo que pertenece a la serie temporal RIVF sería la

diferencia: $82.51 - 77.64 = 4.87$, este procedimiento se repite a lo largo de la señal y se calcula el promedio.

3.2.2.4. Estimación de Frecuencia Respiratoria

El periodo de una señal es el inverso de su frecuencia. En el contexto de la señal PPG y la frecuencia respiratoria, el periodo se refiere al tiempo que tarda en repetirse el patrón asociado con un ciclo respiratorio completo. Si se identifican los picos o valles en la señal PPG que están asociados con la respiración, el periodo entre estos eventos se corresponde con el intervalo de tiempo de un ciclo respiratorio. Si el intervalo entre picos o valles es más corto, significa que hay más eventos (respiraciones) en un período de tiempo determinado, lo que resulta en una frecuencia respiratoria más alta y viceversa (Florentín, 2010).

La frecuencia respiratoria $f_{respiratoria}$ se puede expresar matemáticamente como la inversa del promedio de los intervalos de tiempo entre picos o valles multiplicada por 60 para llevarla a respiraciones por minuto. Si se utiliza los intervalos entre picos ($PP_{intervalos}$), la formula sería:

$$f_{respiratoria} = \frac{60}{promedio(PP_{intervalos})} \quad \dots (22)$$

Donde $promedio(PP_{intervalos})$ representa el valor promedio de los intervalos de tiempo entre picos. Análogamente, si se utiliza los intervalos entre valles $VV_{intervalos}$ la fórmula sería la misma:

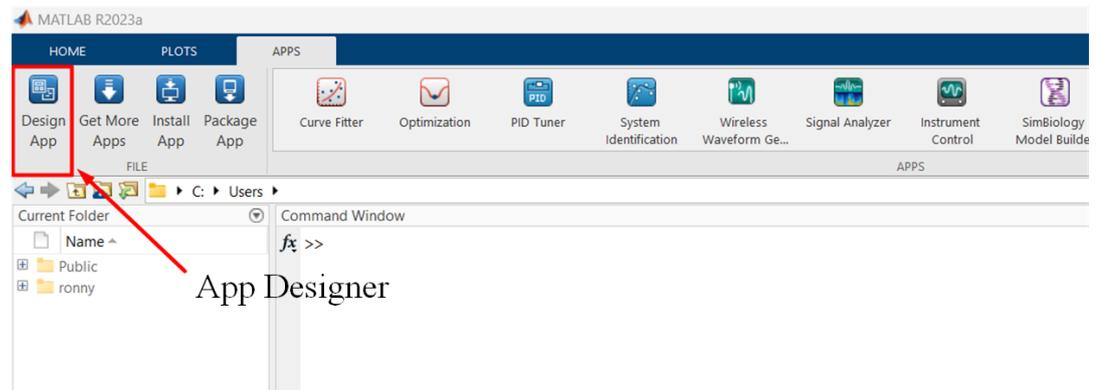
$$f_{respiratoria} = \frac{60}{promedio(VV_{intervalos})} \quad \dots (23)$$

3.2.3. Desarrollo de dashboard.

El desarrollo del dashboard se realizó en MATLAB, utilizando su entorno gráfico App Designer, Para empezar con el diseño se debe seleccionar la pestaña APPS de MATLAB, ahí se encuentra el botón Desing App, como se muestra en la Figura 30.

Figura 30

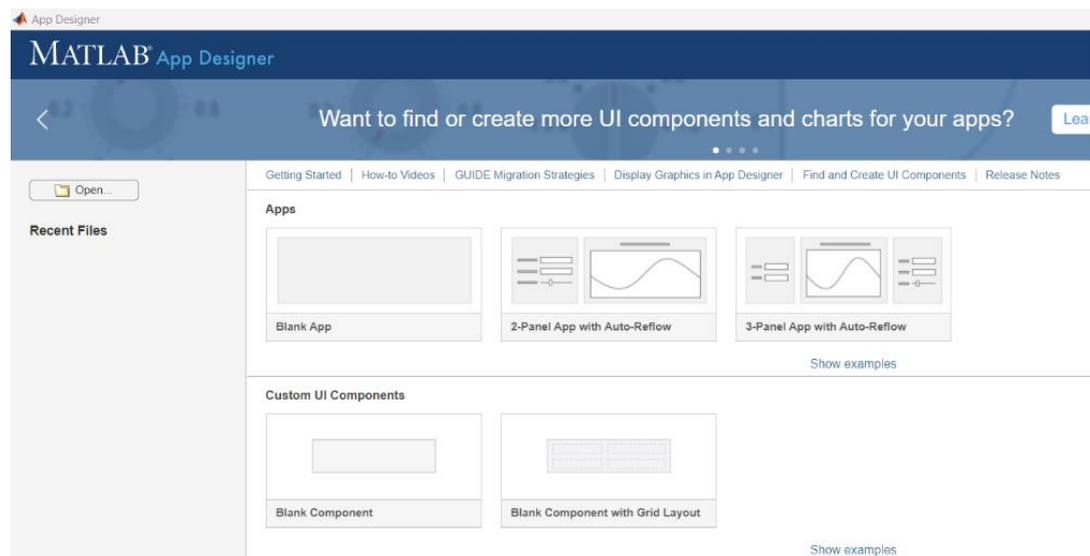
Interfaz gráfica MATLAB botón Desing App



Al hacer clic a este botón, se abrirá la ventana mostrada en la Figura 31 en donde se puede observar diseños de plantillas predefinidas para el desarrollo de Apps.

Figura 31

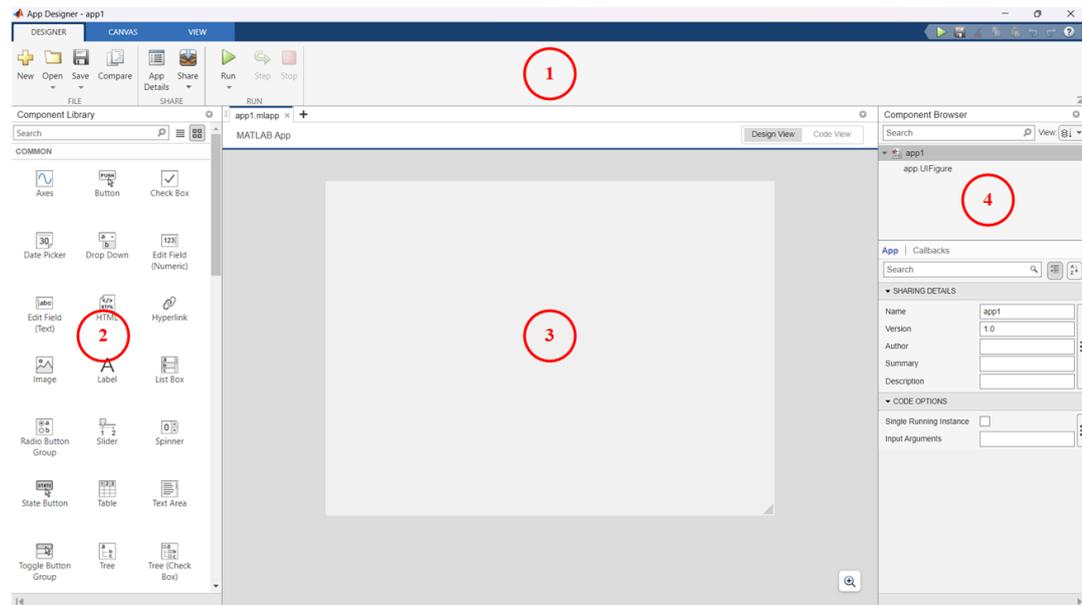
Ventana de entorno de App Designer



En este caso se empieza con una plantilla de aplicación completamente nueva, por lo tanto, se hace clic en “Blank App” para comenzar una aplicación en blanco. A continuación, se abrirá la ventana mostrada en la Figura 32.

Figura 32

Ventana de entorno de App Designer



Nota. En la figura se distinguen 4 secciones principales. 1. Barra de herramientas, 2. Paleta de componentes, 3. Área de Diseño y 4. Navegador de componentes.

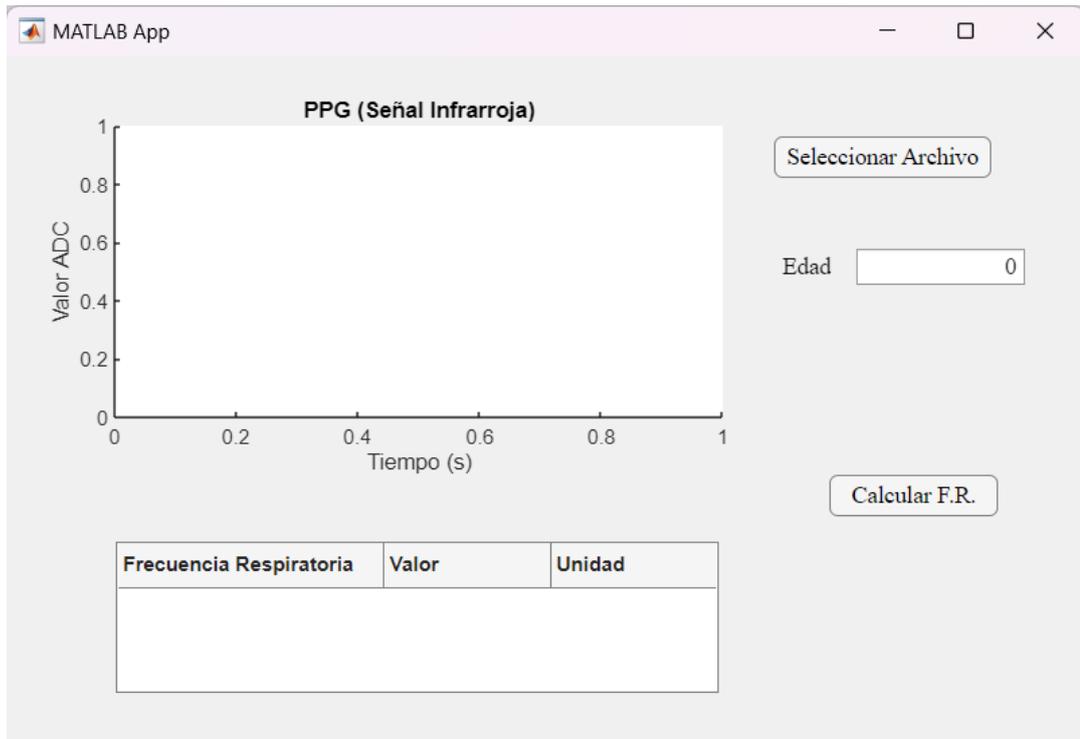
Haciendo uso de los scripts utilizados anteriormente se implementó el diseño del dashboard resultante en cual se ha añadido cinco componentes para su total funcionamiento:

- Axes: permite la visualización grafica de datos.
- Button: permite ejecutar comandos cuando se presiona.
- Edit Field (Numeric): permite la entrada de datos numéricos.
- Table: permite mostrar datos tabulares

La grafica 33 muestra el diseño final del dashboard implementado para el cálculo de la Frecuencia Respiratoria de manera automática, para ello se debe empezar por seleccionar el archivo correspondiente a la data cruda PPG que se quiere analizar, a continuación se debe ingresar la edad de la persona o sujeto a la que le corresponde dicha data, por último se presiona el botón “Calcular F.R.”, en el lado izquierdo superior se graficara la señal PPG del LED infrarrojo y en la parte izquierda inferior se mostraran los resultados obtenidos.

Figura 33

Dashboard para el cálculo de Frecuencia Respiratoria



3.2.4. Validación de prototipo.

La forma de validación estará enfocada a tomar medidas PPG en pacientes y a la vez contar la cantidad de veces que la persona a respirado por minuto de manera manual.

A su vez se calculará el error absoluto y el error relativo con respecto al resultado obtenido. Se utilizarán las siguientes formulas:

$$E_A = |V_E - V_A| \quad \dots (24)$$

$$E_R = \frac{E_A}{V_E} \times 100 \quad \dots (25)$$

Donde:

E_A = Error Absoluto

E_R = Error Relativo

V_E = Valor Exacto

V_A = Valor Aproximado

De igual manera se puede calcular el Error Absoluto Medio (MAE) y el Error Relativo Medio (MRE) para “n” muestras:

$$MAE = \frac{1}{n} \sum_{i=1}^n |V_E - V_A| \quad \dots (26)$$

$$MRE = \frac{1}{n} \sum_{i=1}^n \frac{|V_E - V_A|}{|V_E|} \times 100 \quad \dots (27)$$

3.3. Resultados

3.3.1. Hardware y adquisición de data

Con respecto al prototipo del hardware se logró implementar y tomar las medidas necesarias para el posterior estudio, con la finalidad de una mejor adquisición de señales, se reciclo una carcasa de oxímetro de pulso para colocar el sensor MAX30102 dentro de esta y funcione a modo de pinza. En las Figuras 34 y 35 se observan dos vistas del prototipo final. La Figura 36 muestra la adquisición de la data mediante el prototipo.

Figura 34

Lado inferior del prototipo



Figura 35

Lado frontal y superior del prototipo

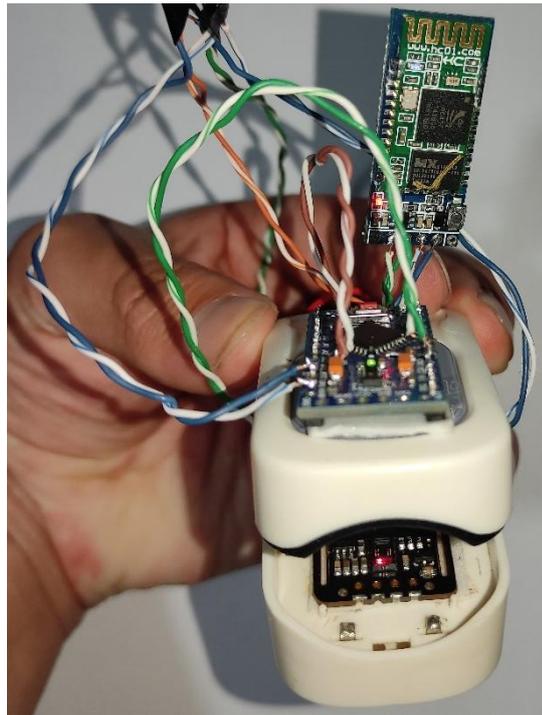
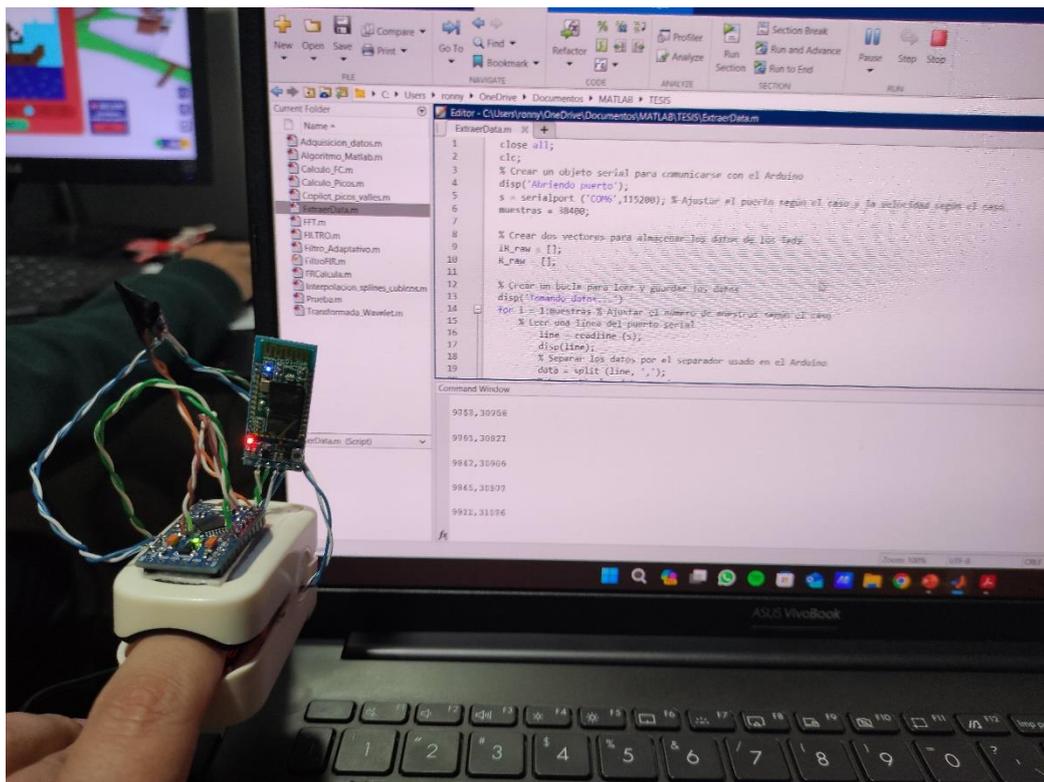


Figura 36

Toma de muestras con el prototipo



3.3.2. Algoritmo, dashboard y validación

A continuación, se muestran los resultados obtenidos por 4 sujetos de diferentes edades en una primera etapa el algoritmo se validó comparando el valor adquirido, con el valor medio en el rango de la frecuencia respiratoria correspondiente a la edad del sujeto.

SUJETO 1

Figura 37

Dashboard del sujeto 1.

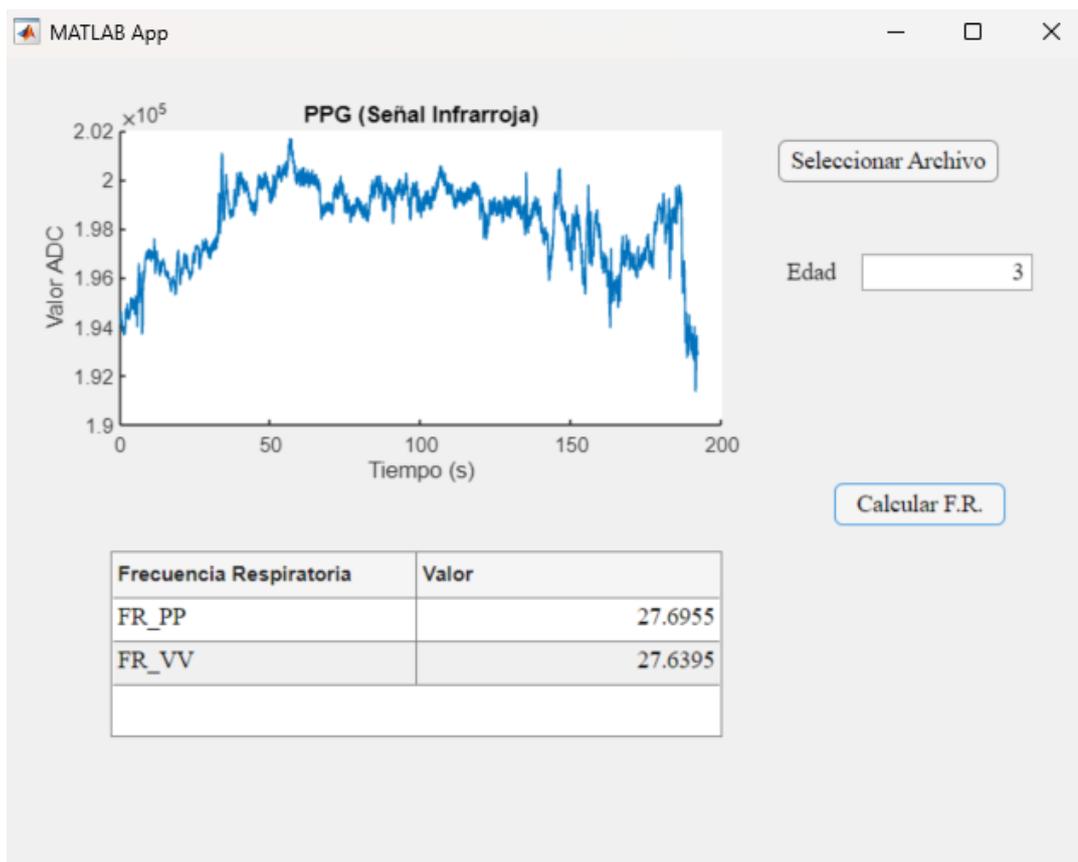


Tabla 13

Resultados obtenidos del sujeto 1.

Edad (años)	Rango	FR (rpm)	EA	ER
03	20 - 35	27.7	0.2	0.73%

SUJETO 2

Figura 38

Dashboard del sujeto 2.

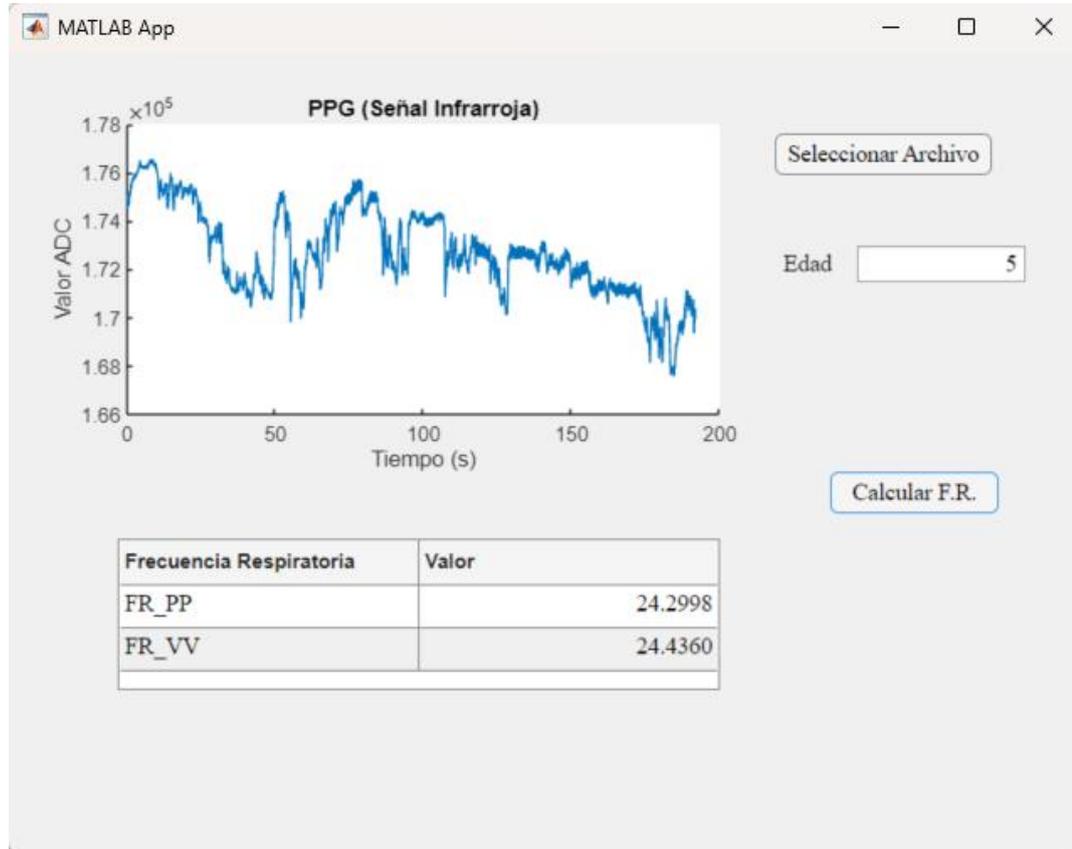


Tabla 14

Resultados obtenidos del sujeto 2.

Edad (años)	Rango	FR (rpm)	EA	ER
05	20 - 30	24.3	0.7	2.8%

SUJETO 3

Figura 39

Dashboard del sujeto 3.

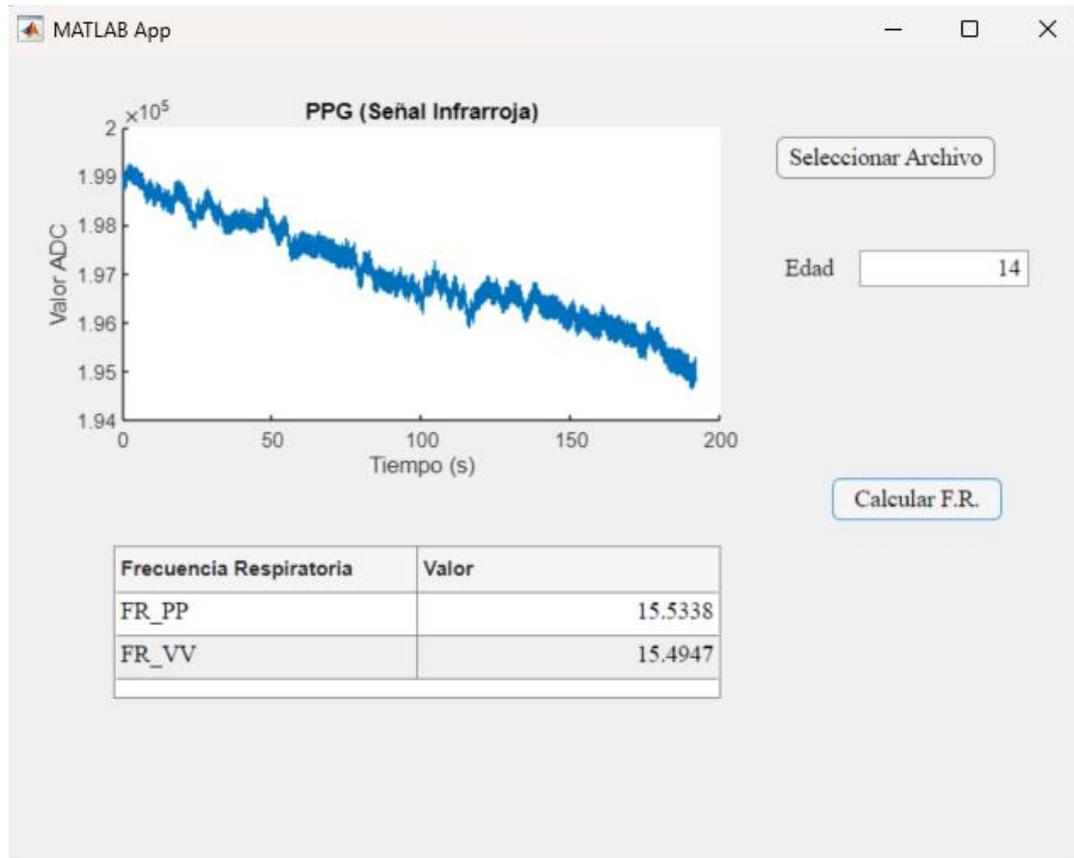


Tabla 15

Resultados obtenidos del sujeto 3.

Edad (años)	Rango	FR (rpm)	EA	ER
14	14 - 20	15.53	1.47	8.6%

SUJETO 4

Figura 40

Dashboard del sujeto 4.

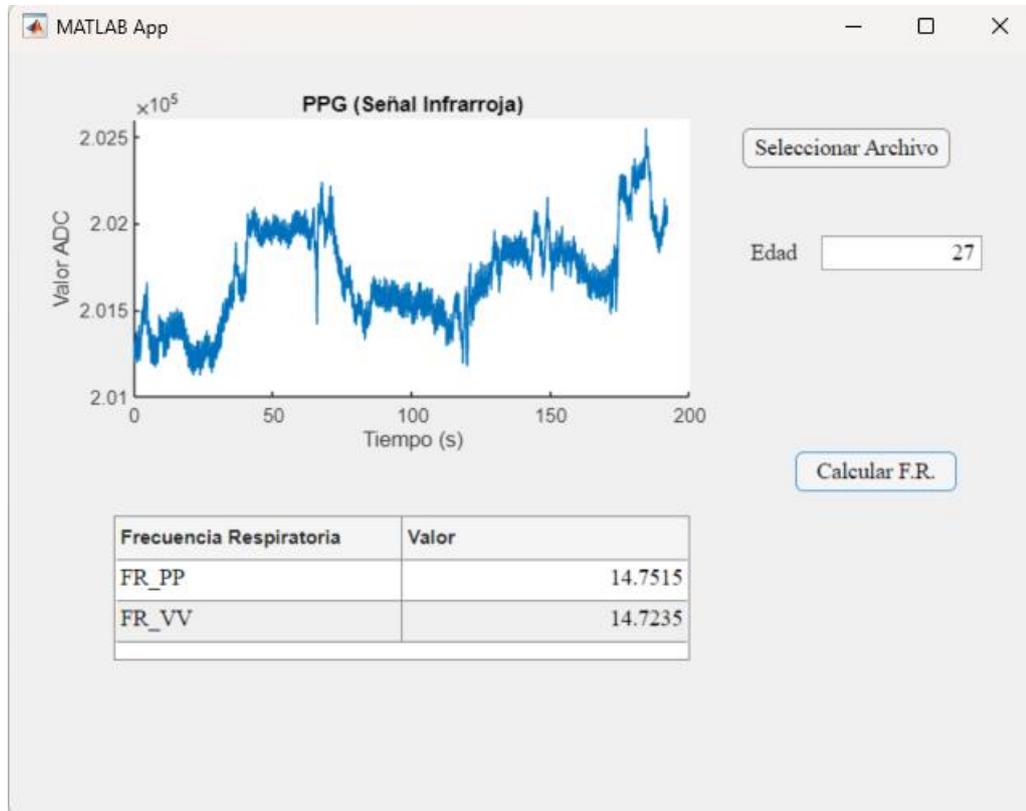


Tabla 16

Resultados obtenidos del sujeto 4.

Edad (años)	Rango	FR (rpm)	EA	ER
27	12 - 20	14.75	1.25	7.8%

La tabla 17, muestra los resultados obtenidos de diferentes sujetos en diferentes rangos de edades.

Tabla 17*Resultados aproximados de estimación de FR*

Edades (años)	Rango de FR (rpm)	Frecuencia Respiratoria (rpm)	Error Absoluto	Error Relativo
03	20 - 35	27.7	0.2	0.73%
05	20 - 30	24.3	0.7	2.8%
05	20 - 30	24.9	0.1	0.4%
14	14 - 20	15.53	1.47	8.6%
27	12 - 20	14.75	1.25	7.8%
31	12 - 20	14.96	1.04	6.5%
31	12 - 20	15.67	0.33	2.1%
31	12 - 20	16.43	0.43	2.69%
31	12 - 20	17.25	1.25	7.8%
72	12 - 16	15.43	1.43	10.2%

Nota. El cálculo se hizo tomando como referencia el valor medio entre el rango de la frecuencia respiratoria.

La Tabla 18, muestra los resultados obtenidos en 20 muestras comparando el valor adquirido mediante el algoritmo con el valor real medido contando las respiraciones de un sujeto de 31 años de edad, durante un minuto.

Tabla 18*Resultados de estimación de FR*

Muestra	Frecuencia Respiratoria Prototipo (rpm)	Frecuencia Respiratoria Real (rpm)	Error Absoluto	Error Relativo
1	16.42	16	0.42	3%
2	16.56	17	0.44	3%
3	16.34	17	0.66	4%
4	16.25	17	0.75	4%
5	15.91	16	0.09	1%
6	16.2	17	0.8	5%

7	16.1	17	0.9	5%
8	16.12	16	0.12	1%
9	15.85	17	1.15	7%
10	16.62	17	0.38	2%
11	16.21	17	0.79	5%
12	15.17	16	0.83	5%
13	15.96	17	1.04	6%
14	16.32	17	0.68	4%
15	16.20	17	0.8	5%
16	16.06	17	0.94	6%
17	16.41	17	0.59	3%
18	16.66	16	0.66	4%
19	16.58	17	0.42	2%
20	16.40	17	0.6	4%

Con las ecuaciones (26) y (27) se calcula el MAE y MRE de las muestras de la tabla 18:

$$MAE = \frac{1}{20} \sum_{i=1}^{20} |V_E - V_A| = 0.653$$

$$MRE = \frac{1}{20} \sum_{i=1}^{20} \frac{|V_E - V_A|}{|V_E|} \times 100 = 4\%$$

IV. DISCUSIÓN DE RESULTADOS

- Como objetivo específico se consideró desarrollar un hardware no invasivo, portable e inalámbrico para la adquisición de señales PPG, estas señales son susceptibles a movimientos intensos, en base a esto, es necesario hacer un procesamiento a las señales obtenidas. La etapa de filtrado de la señal PPG es una de las más importantes, pues esto permite extraer las características de la señal de una forma precisa. En esta investigación se determinó el uso de un filtro Butterworth pasa-banda de orden 3, esto es comparable con los resultados obtenidos por Lizarzaburu y Villanueva (2023) quienes determinaron usar un filtro Butterworth pasa-banda de orden 6 para señales PPG obtenidas de una tarjeta de adquisición de datos.
- El algoritmo propuesto en esta investigación proporciona un MAE más bajo, 0.653 respiraciones por minuto, frente al MAE obtenido con el algoritmo propuesto por M. Pirhonen et al. (2018) de 2.33 respiraciones por minuto y el MAE obtenido con el algoritmo propuesto por Iqbal et al (2022) de 2.05 respiraciones por minuto. Se debe tener en cuenta que el algoritmo propuesto en esta investigación se aplicó a 20 muestras tomadas a través del hardware desarrollado como primer objetivo específico, mientras que las dos últimas investigaciones anteriormente mencionadas aplican su algoritmo sobre un conjunto de datos del Berth Israel Deaconess Medical Centre (BIDMC) que contienen datos de 53 sujetos.

V. CONCLUSIONES

- La principal dificultad para obtener el pulso a partir de las señales PPG son los artefactos de movimiento (MA, por sus siglas en inglés), que alteran fácilmente partes de la señal hasta el punto de que algunos segmentos no permiten identificar la señal de pulso. A pesar de que existen muchas técnicas para reducir su impacto, como los filtros usados en esta investigación, los MA siguen siendo un factor relevante a considerar al confiar en la exactitud de las medidas obtenidas mediante esta técnica en situaciones donde los MA sean intensos
- El algoritmo utilizado tiene un desempeño relativo en las señales evaluadas, sin embargo, se podría hacer un ajuste mayor (un procesamiento más robusto) a la señal PPG para reducir el margen error en la estimación de la frecuencia respiratoria.
- Los resultados obtenidos en la validación avalan que la integración del algoritmo propuesto a un dispositivo de oximetría de pulso disponible comercialmente ampliaría su funcionalidad desde la medición del nivel de saturación de oxígeno y frecuencia cardíaca hasta la medición continua de la frecuencia respiratoria con buena eficiencia en casa y hasta posiblemente en un entorno clínico, después de un estudio más profundo.

VI. REFERENCIAS BIBLIOGRÁFICAS

- Alcorta, N., Pinedo, F. (2021). *Desarrollo de un sistema electrónico biomédico portátil para el monitoreo y registro de signos vitales de las personas en la ciudad de Trujillo – Perú* [Tesis, Universidad Privada Antenor Orrego - UPAO]. <https://hdl.handle.net/20.500.12759/7127>
- Allen J. (2007). *Photoplethysmography and its application in clinical physiological measurement* [Fotopleletismografía y su aplicación en la medición fisiológica clínica]. *Physiological measurement*, 28(3), R1–R39. <https://doi.org/10.1088/0967-3334/28/3/R01>
- Analog Devices. (2019). *MAX30102: High-Sensitivity Pulse Oximeter and Heart-Rate Sensor* [MAX30102: Oxímetro de pulso de alta sensibilidad y sensor de frecuencia cardíaca]. Recuperado de <https://www.analog.com/media/en/technical-documentation/data-sheets/max30102.pdf>
- Arduino Pro Mini 328 - 5V/16MHZ. (s/f). Naylamp Mechatronics - Perú. <https://naylampmechatronics.com/ardusystem-tarjetas/45-pro-mini-328-5v16mhz.html>
- Ballesteros, M., & Torres, R. (2018). *Introducción a los filtros digitales*. http://diegorenza.azurewebsites.net/books/Introduccion_Filtros_Digitales.pdf
- Bian, D., Mehta, P., & Selvaraj, N. (2020). *Respiratory Rate Estimation using PPG: A Deep Learning Approach* [Estimación de la frecuencia respiratoria mediante PPG: un enfoque de aprendizaje profundo]. <https://doi.org/10.1109/embc44109.2020.9176231>
- Charlton, P., Birrenkott, D. A., Bonnici, T., Pimentel, M. A. F., Johnson, A. E. W., Alastruey, J., Tarassenko, L., Watkinson, P. J., Beale, R., & Clifton, D. A. (2018). *Breathing Rate Estimation From the Electrocardiogram and Photoplethysmogram: A Review* [Estimación de la frecuencia respiratoria a partir del electrocardiograma y fotopleletismograma: una revisión]. *IEEE Reviews in Biomedical Engineering*, 11, 2-20. <https://doi.org/10.1109/rbme.2017.2763681>
- Cometti, G. D. & Vogel, F. C. (2018). *Diseño e implementación de un sistema de monitoreo de signos vitales basado en Arduino* [Trabajo final de grado].

- <http://rinfi.fi.mdp.edu.ar/bitstream/handle/123456789/251/GDCometti%2BF CVogel-TFG-IEe-2018.pdf?isAllowed=y&sequence=1>
- Elgendi, M. (2020). *PPG Signal Analysis: An Introduction Using MATLAB®* [Análisis de señales PPG: una introducción al uso de MATLAB®] (1st ed.). CRC Press. <https://doi.org/10.1201/9780429449581>
- Equimed. (2022). *La frecuencia respiratoria: qué es, cómo controlarla y equipos para monitorearla*. Equimed - Equipamiento Médico. <https://equimed.es/la-frecuencia-respiratoria-que-es-como-controlarla-y-equipos-para-monitorearla/>
- Escobar S. (2006). *Diseño de Filtros Digitales [Libro electrónico]*. Facultad de Ingeniería UNAM. <https://odin.fi-b.unam.mx/labdsp/files/libros/filtros.pdf>
- Florentín, D. A. (2010). *Estimación y evaluación de la señal respiratoria a partir de la Señal Fotopletismográfica de Pulso, Señal de Presión Arterial y Señal Electrocardiográfica* [Proyecto fin de carrera]. <https://zaguan.unizar.es/record/4975/files/TAZ-PFC-2010-153.pdf>
- Guardia, A., Chávez, J. (2019). *Aplicación de inteligencia artificial en un sistema de análisis en tiempo real de los signos vitales para pacientes con riesgo cardiaco del hospital Centro Médico Naval* [Tesis, Universidad Ricardo Palma]. <https://hdl.handle.net/20.500.14138/2623>
- Hernández Guevara, L., & Pimentel Hernández, K. S. (2023). *Development of a Modular Prototype of Vital Signs Monitor for Critical Patients* [Desarrollo de un prototipo modular de monitor de signos vitales para pacientes críticos]. TECHNO REVIEW. International Technology, Science and Society Review /Revista Internacional De Tecnología, Ciencia Y Sociedad, 13(3), 1–13. <https://doi.org/10.37467/revtechno.v13.4799>
- Hina, A., & Saadeh, W. (2020). *A Noninvasive Glucose Monitoring SoC Based on Single Wavelength Photoplethysmography* [Un SoC de monitorización de glucosa no invasivo basado en fotopletismografía de longitud de onda única]. *IEEE Transactions on Biomedical Circuits and Systems*, 14(3), 504-515. <https://doi.org/10.1109/tbcas.2020.2979514>
- Iqbal, T., Elahi, A., Ganly, S., Wijns, W., & Shahzad, A. (2022). *Photoplethysmography-Based Respiratory Rate Estimation Algorithm for Health Monitoring Applications* [Algoritmo de estimación de la frecuencia respiratoria basado en fotopletismografía para aplicaciones de seguimiento de

- la salud]. *Journal of medical and biological engineering*, 42(2), 242–252.
<https://doi.org/10.1007/s40846-022-00700-z>
- Jarchi, D., Charlton, P., Pimentel, M. A. F., Casson, A. J., Tarassenko, L., & Clifton, D. A. (2019). *Estimation of respiratory rate from motion contaminated photoplethysmography signals incorporating accelerometry* [Estimación de la frecuencia respiratoria a partir de señales de fotopletismografía contaminadas por movimiento que incorporan acelerometría]. *Healthcare technology letters*, 6(1), 19-26. <https://doi.org/10.1049/htl.2018.5019>
- Katiyar, R. K., Gupta, V., & Pachori, R. B. (2019). *FBSE-EWT-Based Approach for the Determination of Respiratory Rate From PPG Signals* [Enfoque basado en FBSE-EWT para la determinación de la frecuencia respiratoria a partir de señales PPG]. *IEEE sensors letters*, 3(7), 1-4. <https://doi.org/10.1109/lsens.2019.2926834>
- Lizarzaburu, P., Villanueva, J. (2023). *Desarrollo de un filtro digital para señales fotopletismográficas obtenidas de una tarjeta de adquisición de datos en un entorno de laboratorio* [Tesis, Universidad Privada Antenor Orrego - UPAO]. <https://hdl.handle.net/20.500.12759/10140>
- Lucea, C. (2022). *Detección automática de apneas del sueño a partir del electrocardiograma*. E_BUAH. Biblioteca Digital Universidad De Alcalá. <https://ebuah.uah.es/dspace/handle/10017/53497>
- McDuff, D., Gontarek, S., & Picard, R. (2014). *Remote detection of photoplethysmographic systolic and diastolic peaks using a digital camera* [Detección remota de picos sistólicos y diastólicos fotopletismográficos mediante una cámara digital]. *IEEE Transactions on Biomedical Engineering*, 61(12), 2948-2954. <https://doi.org/10.1109/tbme.2014.2340991>
- Ming, D., Sangkaew, S., Chanh, H. Q., Nhat, P. T. H., Yacoub, S., Georgiou, P., & Holmes, A. (2020). *Continuous physiological monitoring using wearable technology to inform individual management of infectious diseases, public health and outbreak responses*. *International Journal of Infectious Diseases*, 96, 648-654. <https://doi.org/10.1016/j.ijid.2020.05.086>
- Oppenheim, A., & Schafer, R. (2011). *Tratamiento de señales en tiempo discreto*. Pearson Educación S. A.
- Pirhonen, M., Peltokangas, M., & Vehkaoja, A. (2018). *Acquiring Respiration Rate from Photoplethysmographic Signal by Recursive Bayesian Tracking of*

- Intrinsic Modes in Time-Frequency Spectra* [Adquisición de la frecuencia respiratoria a partir de una señal fotopleletismográfica mediante seguimiento bayesiano recursivo de modos intrínsecos en espectros de tiempo-frecuencia]. *Sensors*, 18(6), 1693. <https://doi.org/10.3390/s18061693>
- Pollreisz, D., & TaheriNejad, N. (2019). *Detection and Removal of Motion Artifacts in PPG Signals* [Detección y eliminación de artefactos de movimiento en señales PPG]. *Mobile Networks and Applications*, 27(2), 728-738. <https://doi.org/10.1007/s11036-019-01323-6>
- Pulsioxímetro MAX30102. (s/f). Naylamp Mechatronics - Perú. <https://naylampmechatronics.com/biomedico/444-pulsioximetro-max30102.html>
- World Health Organization: WHO & World Health Organization: WHO. (2023). *Enfermedad pulmonar obstructiva crónica (EPOC)*. www.who.int. [https://www.who.int/es/news-room/fact-sheets/detail/chronic-obstructive-pulmonary-disease-\(copd\)](https://www.who.int/es/news-room/fact-sheets/detail/chronic-obstructive-pulmonary-disease-(copd))
- World Health Organization: WHO. (2022). *Neumonía infantil*. www.who.int. <https://www.who.int/es/news-room/fact-sheets/detail/pneumonia>

ANEXOS

Anexo 1. Matriz de Consistencia

Título: Desarrollo de un sistema embebido para la estimación de frecuencia respiratoria basado en señales de fotopleletismografía.

PROBLEMAS	OBJETIVOS	METODOLOGÍA
<p>Problema General</p> <p>La estimación continua de la FR en pacientes de hospitales y clínicas se fundamenta en la utilización de equipos que se manipulan en los centros de atención médica y constituyen dificultades para la supervisión cuando el paciente esta fuera del establecimiento de salud.</p> <p>Problemas Específicos</p> <ul style="list-style-type: none"> • La falta de un dispositivo u oxímetro de pulso que funcione como monitor de signos vitales multipropósito. • La falta de solidez en los métodos existentes para la estimación robusta de FR a partir de señales de PPG. • La necesidad de sistemas que permitan visualizar datos de la FR. • La necesidad de verificar la precisión de los prototipos existentes. 	<p>Objetivo General</p> <p>Desarrollar un sistema embebido para la estimación de frecuencia respiratoria basado en señales de fotopleletismografía.</p> <p>Objetivos Específicos</p> <ul style="list-style-type: none"> • Desarrollar un hardware no invasivo portable, inalámbrico para la adquisición de señales de fotopleletismografía. • Determinar una técnica para mejorar la estimación de la frecuencia respiratoria. • Diseñar un sistema que permita visualizar los datos en un dashboard. • Validar el prototipo mediante pruebas en laboratorio. 	<p>Tipo investigación</p> <p>Aplicada</p> <p>Diseño de investigación</p> <p>Experimental</p> <p>Enfoque de investigación</p> <p>Cuantitativo</p>

Anexo 2. Glosario de términos

Artefactos: Se refiere a una interferencia en una señal originada por alguna fuente ajena al fenómeno que se está analizando.

Dashboard: Se trata de una interfaz de usuario, que puede tener cierta similitud con el tablero de control de un coche, donde se ordena y se muestra la información de una forma que es fácil de entender.

EPOC: Enfermedad pulmonar obstructiva crónica, enfermedad común que reduce el flujo de aire y causa problemas respiratorios.

FC: Frecuencia cardiaca, se refiere al número de veces que el corazón bombea sangre.

Filtro digital: Un filtro digital se basa en operaciones matemáticas que transforman una secuencia de números (datos de entrada) en otra secuencia de números (datos de salida).

Fotopletismografía: Es una técnica de pletismografía en la cual se utiliza un haz de luz para determinar el volumen de un órgano.

FR: Frecuencia respiratoria, es la cantidad de respiraciones por minuto que da una persona.

IRA: Infección respiratoria aguda, es una enfermedad que se produce en el aparato respiratorio, y son causadas por diferentes microorganismos como virus y bacterias.

Microprocesador: Es un dispositivo que realiza funciones de la CPUU en un único circuito integrado.

OMS: Organización mundial de la Salud, es el organismo especializado de la Organización de las Naciones Unidas (ONU) que tiene como labor gestionar políticas de prevención, promoción e intervención a nivel mundial a favor de la salud.

Sistema embebido: Un sistema embebido es un sistema de computación que se apoya en un microprocesador o un microcontrolador y que se especializa en hacer una o unas pocas actividades concretas, usualmente en un sistema de computación que trabaja en tiempo real.

Anexo 3. Código en Arduino para la comunicación y configuración del sensor

MAX30102

```
#include <Wire.h>
#include "MAX30105.h"
#include <SoftwareSerial.h>

SoftwareSerial bluetoothSerial(2, 3); // RX, TX

MAX30105 particleSensor; // Inicializar MAX30102 con I2C

void setup()
{
  Serial.begin(115200);
  bluetoothSerial.begin(115200); // Iniciar la comunicación serie con el
  módulo Bluetooth

  // Inicializar el sensor
  if (particleSensor.begin(Wire, I2C_SPEED_FAST) == false) // Utilice el
  puerto I2C predeterminado, velocidad de 400 Hz
  {
    Serial.println("MAX30102 no fue encontrado. Por favor revise el
  cableado/alimentación. ");
    while (1);
  }

  byte ledBrightness = 250; // Opciones: 0=apagado a 255=50mA
  byte sampleAverage = 1; // Opciones: 1, 2, 4, 8, 16, 32
  byte ledMode = 2; // Opciones: 1 = Solo Red, 2 = Red + IR, 3 = Red + IR +
  Green
  int sampleRate = 400; // Opciones: 50, 100, 200, 400, 800, 1000, 1600, 3200
  int pulseWidth = 69; // Opciones: 69, 118, 215, 411 microsegundos
  int adcRange = 16384; // Opciones: 2048, 4096, 8192, 16384 nA

  particleSensor.setup(ledBrightness, sampleAverage, ledMode, sampleRate,
  pulseWidth, adcRange); // Configure el sensor con estos ajustes
}

void loop() {
  particleSensor.check(); // Comprobar el sensor
  while (particleSensor.available()) {
    // Leer IR almacenado
    int irValue = particleSensor.getIR();
    // Leer red almacenado
    int redValue = particleSensor.getRed();
  }
}
```

```
// Enviar datos por Bluetooth
bluetoothSerial.print(irValue);
bluetoothSerial.print(",");
bluetoothSerial.println(redValue);
// Leer el siguiente conjunto de muestras
particleSensor.nextSample();
}
}
```

Anexo 4. Código en MATLAB para la lectura y representación de datos del sensor

```
close all;
clc;
% Crear un objeto serial para comunicarse con el Arduino
disp('Abriendo puerto');
s = serialport ('COM5',115200); % Ajustar el puerto según el caso y la velocidad
según el caso
muestras = 38400;

% Crear dos vectores para almacenar los datos de los leds
IR_raw = [];
R_raw = [];

% Crear un bucle para leer y guardar los datos
disp('Tomando datos...')
for i = 1:muestras % Ajustar el número de muestras según el caso
    % Leer una línea del puerto serial
    line = readline (s);
    disp(line);
    % Separar los datos por el separador usado en el Arduino
    data = split (line, ',');
    % Convertir los datos a números
    data = str2double (data);
    % Guardar los datos en los vectores correspondientes
    IR_raw (i) = data(1);
    R_raw (i) = data(2);
end

% Cerrar el puerto serial
delete (s);
clear s;
disp('Cerrando puerto')

% Obtener la fecha y hora actual como un número
fecha = datetime("now");
% Convertir la fecha y hora a una cadena con el formato 'yyyy-mm-dd_HH-MM-SS'
fecha_str = string(fecha, 'yyyyMMddHHmmSS');

% Crear el nombre del archivo
filename = [strcat('C:\Users\ronny\OneDrive\Escritorio\Data_MATLAB\', fecha_str,
'.mat')];
% Guardar los vectores en el archivo con el nombre generado
save (filename, 'IR_raw', 'R_raw');

% Graficar los datos de los sensores
plot (IR_raw(22:length(IR_raw)), 'b'); % Graficar el primer sensor en azul
hold on; % Mantener el gráfico
plot (R_raw(22:length(R_raw)), 'r'); % Graficar el segundo sensor en rojo
hold off; % Terminar el gráfico
xlabel ('Muestra'); % Etiquetar el eje x
ylabel ('Valor'); % Etiquetar el eje y
legend ('Sensor Infrarrojo', 'Sensor rojo'); % Añadir una leyenda
```

Anexo 5. Código en MATLAB para filtrar la señal PPG

```
% Filtro Butterworth
Fs=200; % Frecuencia de muestreo del sensor
n=3; % Orden del filtro
wn = [0.2 0.33]/(Fs/2); % Puntos de corte
[num,den]=butter(n,wn,'Bandpass'); % num y den son los parámetros del filtro
freqz(num,den,[],Fs); % Respuesta del filtro en frecuencia

R_filtrada = filter(num,den,R_raw); % Señal Roja filtrada
R = R_filtrada(1:length(R_filtrada)); % Señal Roja filtrada recortada

IR_filtrada = filter(num,den,IR_raw); % Señal infrarroja filtrada
IR = IR_filtrada(1:length(IR_filtrada)); % Señal Roja filtrada recortada

figure('name','Filtro');
subplot(2,1,1);
plot(R,'r');
title('Señal roja')

subplot(2,1,2);
plot(IR,'b');
title('Señal infarroja')
```

Anexo 6. Código en MATLAB para la ubicación de picos y valles

```
%% Picos y Valles
fs = 200; % Frecuencia de muestreo
tm = 1/fs; % Tiempo de muestra
t = tm.* (1:length(R));
signal_PPG = R;

% Configurar los parámetros para la función findpeaks
up = 5; % Umbral de prominencia para identificar picos y valles
dm_p = 100; % Distancia mínima entre picos y valles

% Encontrar los picos en la señal PPG
[p, ubi_p] = findpeaks(signal_PPG, 'MinPeakProminence', up, 'MinPeakDistance',
dm_p);

% Invertir la señal para encontrar los valles
signal_PPG_invertida = -signal_PPG;

% Encontrar los valles en la señal PPG invertida (equivalente a encontrar picos en
la señal original)
[v, ubi_v] = findpeaks(signal_PPG_invertida, 'MinPeakProminence', up,
'MinPeakDistance', dm_p);

% Invertir los valores de los valles
v = -v;

% Obtener los tiempos correspondientes a las ubicaciones de los picos y valles
tp = t(ubi_p);
tv = t(ubi_v);

% Visualizar la señal y marcar los picos y valles
figure;
plot(t, signal_PPG);
hold on;
plot(tp, p, 'ro', 'MarkerSize', 10, 'DisplayName', 'Picos');
plot(tv, v, 'bo', 'MarkerSize', 10, 'DisplayName', 'Valles');
title('Señal PPG con Picos y Valles Identificados');
xlabel('Tiempo');
ylabel('Amplitud');
legend;

% Definir la carpeta donde se guardará el archivo .mat
ruta_carpeta = 'C:\Users\ronny\OneDrive\Escritorio\Resultados\';

% Generar un nombre de archivo con la fecha y hora actual
fecha_hora_actual = datetime('now', 'Format', 'yyyyMMdd_HHmss');
nombre_archivo = ['datos_ppg_', char(fecha_hora_actual), '.mat'];

% Guardar los datos en el archivo .mat en la carpeta especificada
ruta_archivo_completa = fullfile(ruta_carpeta, nombre_archivo);
save(ruta_archivo_completa, 'signal_PPG', 'tp', 'tv', 'intervalos_PP',
'intervalos_VV', 'FR_PP', 'FR_VV');
```

Anexo 7. Código en MATLAB para el cálculo de la Frecuencia Respiratoria

```
%%Frecuencia Respiratoria
% Cálculo de intervalos R-R o V-V
intervalos_PP = diff(tp);
intervalos_VV = diff(tv);

% Cálculo de la frecuencia respiratoria
FR_PP = 60 / (mean(intervalos_PP));
FR_VV = 60 / (mean(intervalos_VV));

% Mostrar los resultados
disp(['Frecuencia Respiratoria (por picos): ', num2str(FR_PP), ' respiraciones por
minuto']);
disp(['Frecuencia Respiratoria (por valles): ', num2str(FR_VV), ' respiraciones
por minuto']);
```

Anexo 8. Código en App Designer de MATLAB para el desarrollo del Dashboard

```
classdef Estimacion_FR < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                matlab.ui.Figure
        CalcularFRButton        matlab.ui.control.Button
        ResultadosTable         matlab.ui.control.Table
        EdadEditField           matlab.ui.control.NumericEditField
        EdadEditFieldLabel     matlab.ui.control.Label
        SeleccionarArchivoButton matlab.ui.control.Button
        UIAxes                  matlab.ui.control.UIAxes
    end

    properties (Access = private)
        SelectedFilePath % Description
        AgeEdit % Description
    end

    % Callbacks that handle component events
    methods (Access = private)

        % Button pushed function: SeleccionarArchivoButton
        function SeleccionarArchivoButtonPushed(app, event)
            [filename, path] = uigetfile({'*.xlsx'}, 'Selecciona un archivo
Excel');
            if filename ~= 0 % Si el usuario no cancela la selección
                % Actualiza las propiedades de la App Designer con la información
del archivo seleccionado
                app.SelectedFilePath = fullfile(path, filename);
                disp(['Archivo seleccionado: ', app.SelectedFilePath]);
            end
        end

        % Button pushed function: CalcularFRButton
        function CalcularFRButtonPushed(app, event)
            if isempty(app.SelectedFilePath)
                disp('Por favor, selecciona un archivo antes de presionar este
botón.');
```

```

%% Frecuencia de Muestreo
Fs =200;

%% Longitud de la señal
N = length(IR_raw);

%% Vector de tiempo en segundos
t1 = (0:N-1)/Fs;

%% Gráfico de IR_raw
axes(app.UIAxes); % Establece el área de gráficos
plot(app.UIAxes,t1,IR_raw);

%% Definir los puntos de corte según la edad
if age >= 1 && age <= 3
    wn = [0.33 0.58]/(Fs/2);
elseif age > 3 && age <= 5
    wn = [0.33 0.5]/(Fs/2);
elseif age > 5 && age <= 12
    wn = [0.25 0.5]/(Fs/2);
elseif age > 12 && age <= 18
    wn = [0.23 0.33]/(Fs/2);
elseif age > 18 && age <= 70
    wn = [0.2 0.33]/(Fs/2);
else
    wn = [0.2 0.27]/(Fs/2);
end
%% Aplicar el filtro Butterworth
n = 3; % Orden del filtro
[num, den] = butter(n, wn, 'Bandpass');

%% Filtrar la señal
IR_filtrada = filter(num, den, IR_raw);
IR = IR_filtrada(1:length(IR_filtrada));

tm = 1/Fs; % Tiempo de muestra
t = tm .* (1:length(IR));
signal_PPG = IR;
% signal_PPG = abs(hilbert(R));

%% Configurar los parámetros para la función findpeaks
up = 2; % Umbral de prominencia para identificar picos y valles
dm_p = 100; % Distancia mínima entre picos y valles

%% Encontrar los picos en la señal PPG
[p, ubi_p] = findpeaks(signal_PPG, 'MinPeakProminence', up,
'MinPeakDistance', dm_p);

%% Invertir la señal para encontrar los valles
signal_PPG_invertida = -signal_PPG;

%% Encontrar los valles en la señal PPG invertida
[v, ubi_v] = findpeaks(signal_PPG_invertida, 'MinPeakProminence', up,
'MinPeakDistance', dm_p);

%% Invertir los valores de los valles
v = -v;

```

```

y valles % Obtener los tiempos correspondientes a las ubicaciones de los picos

tp = t(ubi_p);
tv = t(ubi_v);

% Cálculo de intervalos R-R o V-V
intervalos_PP = diff(tp);
intervalos_VV = diff(tv);

% Cálculo de la frecuencia respiratoria
FR_PP = 60 / (mean(intervalos_PP));
FR_VV = 60 / (mean(intervalos_VV));

% Mostrar los resultados
disp(['Frecuencia Respiratoria (por picos): ', num2str(FR_PP), '
respiraciones por minuto']);
disp(['Frecuencia Respiratoria (por valles): ', num2str(FR_VV), '
respiraciones por minuto']);

% Actualizar el UITable
app.ResultadosTable.Data = {'FR_PP', FR_PP, 'RPM'; 'FR_VV', FR_VV,
'RPM'};

% Definir la carpeta donde se guardará el archivo .mat
ruta_carpeta = 'C:\Users\ronny\OneDrive\Escritorio\Resultados\';

% Generar un nombre de archivo con la fecha y hora actual
fecha_hora_actual = datetime('now', 'Format', 'yyyyMMdd_HHmss');
nombre_archivo = ['datos_ppg_', char(fecha_hora_actual), '.mat'];

% Guardar los datos en el archivo .mat en la carpeta especificada
ruta_archivo_completa = fullfile(ruta_carpeta, nombre_archivo);
save(ruta_archivo_completa, 'signal_PPG', 'tp', 'tv', 'intervalos_PP',
'intervalos_VV', 'FR_PP', 'FR_VV');

end

% Button down function: UIAxes
function UIAxesButtonDown(app, event)

end

end

% Component initialization
methods (Access = private)

% Create UIFigure and components
function createComponents(app)

% Create UIFigure and hide until all components are created
app.UIFigure = uifigure('Visible', 'off');
app.UIFigure.Position = [100 100 640 480];
app.UIFigure.Name = 'MATLAB App';

% Create UIAxes
app.UIAxes = uiaxes(app.UIFigure);
title(app.UIAxes, 'PPG (Señal Infrarroja)')

```

```

xlabel(app.UIAxes, 'Tiempo (s)')
ylabel(app.UIAxes, 'Valor ADC')
zlabel(app.UIAxes, 'Z')
app.UIAxes.ButtonDownFcn = createCallbackFcn(app, @UIAxesButtonDown,
true);

app.UIAxes.Position = [25 230 409 228];

% Create SeleccionarArchivoButton
app.SeleccionarArchivoButton = uibutton(app.UIFigure, 'push');
app.SeleccionarArchivoButton.ButtonPushedFcn = createCallbackFcn(app,
@SeleccionarArchivoButtonPushed, true);
app.SeleccionarArchivoButton.FontName = 'Times New Roman';
app.SeleccionarArchivoButton.FontSize = 14;
app.SeleccionarArchivoButton.Position = [456 408 129 25];
app.SeleccionarArchivoButton.Text = 'Seleccionar Archivo';

% Create EdadEditFieldLabel
app.EdadEditFieldLabel = uilabel(app.UIFigure);
app.EdadEditFieldLabel.HorizontalAlignment = 'right';
app.EdadEditFieldLabel.FontName = 'Times New Roman';
app.EdadEditFieldLabel.FontSize = 14;
app.EdadEditFieldLabel.Position = [456 344 34 22];
app.EdadEditFieldLabel.Text = 'Edad';

% Create EdadEditField
app.EdadEditField = uieditfield(app.UIFigure, 'numeric');
app.EdadEditField.ValueDisplayFormat = '%.0f';
app.EdadEditField.Tag = 'EdadEditField';
app.EdadEditField.FontName = 'Times New Roman';
app.EdadEditField.FontSize = 14;
app.EdadEditField.Position = [505 344 100 22];

% Create ResultadosTable
app.ResultadosTable = uitable(app.UIFigure);
app.ResultadosTable.ColumnName = {'Frecuencia Respiratoria'; 'Valor';
'Unidad'};
app.ResultadosTable.RowName = {};
app.ResultadosTable.FontName = 'Times New Roman';
app.ResultadosTable.FontSize = 14;
app.ResultadosTable.Position = [66 100 357 91];

% Create CalcularFRButton
app.CalcularFRButton = uibutton(app.UIFigure, 'push');
app.CalcularFRButton.ButtonPushedFcn = createCallbackFcn(app,
@CalcularFRButtonPushed, true);
app.CalcularFRButton.FontName = 'Times New Roman';
app.CalcularFRButton.FontSize = 14;
app.CalcularFRButton.Position = [489 206 100 25];
app.CalcularFRButton.Text = 'Calcular F.R.';

% Show the figure after all components are created
app.UIFigure.Visible = 'on';
end
end

% App creation and deletion
methods (Access = public)

```

```
% Construct app
function app = Estimacion_FR

    % Create UIFigure and components
    createComponents(app)

    % Register the app with App Designer
    registerApp(app, app.UIFigure)

    if nargin == 0
        clear app
    end
end

% Code that executes before app deletion
function delete(app)

    % Delete UIFigure when app is deleted
    delete(app.UIFigure)
end
end
end
```